
TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky a mezioborových studií

Studijní program: B2612 – Elektrotechnika a informatika

Studijní obor: 2612R011 / Elektronické informační a řídicí systémy

Bezdrátový přenos signálu v reálné aplikaci na letadle

Wireless Communication for Real Application on Airplane

Bakalářská práce

Autor: **Jakub Nečásek**

Vedoucí práce: Ing. Lukáš Hubka, Ph.D.

Konzultant: Ing. Petr Školník, Ph.D.

V Liberci 18. 5. 2011





Prohlášení

Byl(a) jsem seznámen(a) s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 o právu autorském, zejména § 60 (školní dílo).

Beru na vědomí, že TUL má právo na uzavření licenční smlouvy o užití mé bakalářské práce a prohlašuji, že **s o u h l a s í m** s případným užitím mé bakalářské práce (prodej, zapůjčení apod.).

Jsem si vědom(a) toho, že užít své bakalářské práce či poskytnout licenci k jejímu využití mohu jen se souhlasem TUL, která má právo ode mne požadovat přiměřený příspěvek na úhradu nákladů, vynaložených univerzitou na vytvoření díla (až do jejich skutečné výše).

Bakalářskou práci jsem vypracoval(a) samostatně s použitím uvedené literatury a na základě konzultací s vedoucím bakalářské práce a konzultantem.

Datum

Podpis



Abstrakt

V úvodu bakalářská práce pojednává o možných řešeních bezdrátové komunikace pro výzkumné a průmyslové použití. Hodnotí také jejich možnou aplikaci pro prostředí laboratorní úlohy „Letadlo“ v učebně TK4. V následující části práce jsou nabízena možná řešení modulu pro bezdrátové odměřování. Dále je v samostatné kapitole popsán vývoj a funkce konkrétního řešení odměřování pro rotující část úlohy. Zde se zabývá jeho důležitými konstrukčními prvky, jimiž jsou mikrokontroler, komunikační modul a snímače zrychlení. Tento modul pracuje jako převodník naměřených hodnot ze snímačů na data pro bezdrátovou komunikaci. Vysvětlena je též konstrukce a účel vyhodnocujícího zapojení na principu stejného řídicího a komunikačního obvodu a dále digitálně-analogového převodníku a operačního zesilovače. Ve svém principu převodník signálu ze snímačů provádí úpravu dat z bezdrátové komunikace na tvar zpracovatelný digitálně-analogovými převodníky. Problémy vzniklé při konstrukci a dokončení modulů pro odměřování jsou zde popsány i s řešeními. Vysvětlen je i postup zabudovávání hotového odměřování do úlohy. V další kapitole práce je popsán program použitý pro řízení mikrokontrolerů a je zde diskutováno i jeho možné vylepšení. Dále jsou zde uvedeny výsledky měření na hotové úloze s nastíněním možného pokračování v tomto tématu. Také jsou zde uvedeny další aplikace, do kterých může být zapojení nasazeno. V obsahu příloh je také úplná dokumentace pro výrobu modulů, kompletní výpisy zdrojových kódů s anglickými popiskami a grafy z naměřených dat.

Klíčová slova: Letadlo, PIC24F04KA201, Bluetooth, Odměřování, Bezdrátová komunikace.



Abstract

At the beginning bachelor thesis deals with possible solutions for wireless communications for research and industrial applications. It also assesses their potential application for laboratory instrument called "Airplane" in the TK4 classroom. In the following part of this bachelor thesis are offered possible solutions for wireless measuring module. Then a separate chapter describes the development and function of a specific solution for measuring on the rotating part of the laboratory instrument. This part also deals with its major structural elements, which are the microcontroller, a communication module and inertial sensors. This module works as a converter reading values from the sensors and converting them to the data for wireless communication. In the next part of report is explained the structure and purpose of analyzing converter involving the same principle of control and communication circuits and explain function of digital-to-analog converters and operational amplifiers. Operational principle is based on data modification that comes from wireless communications to the shape that digital-to-analog converters are processing. Problems arising from the construction and completion of measuring modules are described here with their solutions. There is also explanation of procedures performed to install modules to the laboratory instrument. The next chapter describes the program used for the control of microcontrollers and also features the potential improvements. Next in Bachelor thesis are the results of measurements on laboratory instrument and some of a possible continuation of this topic. Also listed are other applications in which can be wireless module deployed. The content of the annex is also complete documentation for the production of modules, complete source code listings with English labels and the diagrams of measured data.

Key words: Airplane, PIC24F04KA201, Bluetooth, Admeasure, Wireless communication.



Obsah

Prohlášení	3
Abstrakt	4
Abstract	5
Seznam obrázků	8
1 Úvod	9
2 Bezdrátová komunikace	11
2.1 Pásmo 433,92 a 868 MHz	11
2.2 IEEE802.15.4 a nadstavba ZigBee	12
2.3 IEEE802.11 Wi-Fi	14
2.4 IEEE802.15.1 Bluetooth	14
3 Možné návrhy řešení bezdrátového odměřování	17
3.1 Freescale ZSTAR	17
3.2 Použití modulů 433/868 MHz	18
3.3 Wi-Fi měřicí úloha	19
3.4 Vývojové ZigBee moduly RC220x	19
4 Popis vybraného řešení – modul pro letadlo	21
4.1 Mikrokontroler	21
4.2 Bluetooth transceiver	23
4.3 Akcelerometr	24
4.4 Gyroskop	25
4.5 Popis vývoje a funkce zapojení modulu pro letadlo	26
4.6 Instalace modulu do makety letadla	33
5 Popis vybraného řešení – modul pro převod dat	37
5.1 Digitálně-analogový převodník	37
5.2 Operační zesilovač	38
5.3 Popis vývoje a funkce zapojení převodníku pro měřicí kartu	39
6 Instalace modulů do učebny	43
6.1 Instalace převodníku pro měřicí kartu	43
6.2 Instalace letadla na závěs	44



7	Vývoj a popis programu mikrokontroleru.....	45
7.1	Program pro modul v letadle	46
7.2	Program pro převodník u PC karty.....	48
7.3	Rychlost měření a vyhodnocení dat	50
8	Měření na hotové úloze	51
9	Závěr.....	52
	Seznam použité literatury.....	53
	Seznam příloh na CD	55
	Příloha A	56
	Příloha B.....	57
	Příloha C.....	58
	Příloha D	59
	Příloha E.....	60
	Příloha F	63
	Příloha G	67
	Příloha H	68
	Příloha I.....	69



Seznam obrázků

Obr. 1.1: Laboratorní úloha zavěšená pod stropem.....	9
Obr. 2.1: Topologie sítě ZigBee.....	13
Obr. 3.1: Moduly ZSTAR (www.freescale.com).....	17
Obr. 3.2: Blokové schéma s 433/868 MHz moduly.....	18
Obr. 4.1: Datový paket komunikace UART.....	23
Obr. 4.2: Přední strana Bluetooth modulu.....	24
Obr. 4.3: Zadní strana Bluetooth modulu.....	24
Obr. 4.4: Soustava snímání zrychlení.....	25
Obr. 4.5: Schéma převodníku RS232 – UART.....	27
Obr. 4.6: Deska plošných spojů i se součástkami pro převodník.....	28
Obr. 4.7: Hotový převodník.....	28
Obr. 4.8: Schéma zapojení modulu pro letadlo.....	31
Obr. 4.9: Spodní strana desky pro letadlo.....	32
Obr. 4.10: Vrchní strana desky pro letadlo.....	32
Obr. 4.11: Hotové zapouzdření měřicího modulu před instalací do letadla.....	34
Obr. 4.12: Vyříznutý otvor v letadle pro zasunutí krabičky.....	35
Obr. 5.1: SPI komunikace s převodníkem.....	38
Obr. 5.2: První část schématu převodníku pro měřicí kartu.....	40
Obr. 5.3: Druhá část schématu převodníku pro měřicí kartu.....	41
Obr. 5.4: Spodní strana desky pro převodník.....	42
Obr. 5.5: Vrchní strana desky pro převodník.....	42
Obr. 6.1: Zabudované zapojení převodníku pro měřicí kartu.....	43
Obr. 6.2: Souřadný systém letadla.....	44
Obr. 7.1: Testovací zapojení na nepájivém poli.....	45
Obr. 7.2: Vývojový diagram programu pro letadlo.....	47
Obr. 7.3: Vývojový diagram obsluhy přerušení u převodníku pro kartu.....	48
Obr. 7.4: Obsluha přerušení a hlavní program převodníku pro kartu.....	49



1 Úvod

Dominantou učebny TK4 na budově A Technické univerzity v Liberci, je laboratorní úloha „Letadlo“. Tato úloha je zavěšena pod stropem na konci učebny, jak je vidět na obr. 1.1



Obr. 1.1: Laboratorní úloha zavěšená pod stropem

Jak je na první pohled patrné, sestává z polystyrénového modelu stíhacího letadla a jeho závěsu. Po přivedení napětí na motor se model rozpohybuje a působením odstředivé síly je vychýlen od svislé osy rotace. Dlouhým závěsem a váhou makety letadla s motorem je dána dynamika, kterou je právě tato regulovaná soustava zajímavá. I když by to vypadalo, že maketa letadla zde plní jen funkci dekorační, není tomu tak. Tělo modelu klade při pohybu vzduchem odpor a tím přispívá k nelineárnímu průběhu vyosení na výkonu motoru. Dále už z principu, že se jedná o model letadla, jeho vodorovné a svislé plochy směrově stabilizují pohyb. Co však na první pohled patrné není, ačkoli se jedná o část úlohy vcelku základní, je její srdce – tedy centrum řízení a odměřování signálu. Toto je uloženo v instalační krabici pod deskou stolu. Obsahuje výkonný spínaný zdroj z počítače, který slouží jako zdroj napětí 9V pro motor letadla. Ten je sice jen 6V, ale z důvodu dlouhého vedení a úbytkům na něm se napětí ze zdroje dostane na správnou úroveň. Zdroj také napájí převodník analogového napětí 0-10V na PWM signál s mikrokontrolerem PIC. Toto zapojení obsahuje i výkonový člen, který napětí ze zdroje spíná do motoru v závislosti na vstupním analogovém signálu. Závěs obsahuje snímač natočení letadla od svislé osy. Jde o bezkontaktní snímač na principu Hallova jevu. Pro svoji práci vyžaduje napájení 5 V a jeho výstupním signálem je napětí



v rozsahu 0 – 5 V. Napájení snímače a jeho výstupní signál jsou vedeny v liště podél silových vodičů pomocí koaxiální dvojlinky. Toto vedení má přibližně 15m a právě díky přítomnosti silových vodičů je výsledný signál značně zarušený. Také přenos pomocí kartáčů z rotační části je velmi rušivým elementem. Proto hlavním cílem této bakalářské práce je nahrazení metalického vedení a kroužků v cestě signálu bezdrátovým přenosem. Pro detaily mechanické konstrukce je možno nahlédnout do projektu z roku 2010, vypracovaného za účelem rekonstrukce této úlohy.

Jednoduché měření odklonění od svislé osy je pro základní regulaci dostačující, avšak není to příliš zajímavé pro výuku. Pro učinění této úlohy více atraktivní pro studenty i zájemce při dnech otevřených dveří, by bylo vhodné úlohu obohatit o moderní a v dnešní době i módní snímače zrychlení. Přidání dalších snímačů bude znamenat větší nároky na přenos dat, aby byla zajištěna zdánlivá spojitost odměřování. Celá úloha má však velmi pomalou dynamiku, tedy datové toky nebudou kritické. Dále pro budoucí využití je výhodné mít na úloze připraveno několik volných signálů ovládaných přímo počítačem.

Nutnou podmínkou k odměřování je také jeho plná kompatibilita s prostředím programu Matlab a jeho měřicí kartou. Jedná se o unifikovaný signál 0 – 10 V, který karta podporuje. Je tomu tak hlavně z důvodu již existujícího ovládání motoru pomocí této měřicí karty a programu Matlab.



2 Bezdrátová komunikace

V současnosti je bezdrátový přenos informací na rádiových vlnách všude kolem nás. Mnozí by si život bez něj ani nedokázali představit. Rozhlas, televize, telefon, internetové připojení, ale také rádiová komunikace letadel, lodí atd. to vše je dnes každodenně přítomno okolo nás „ve vzduchu“. A protože je člověk tvor zvědavý, zkouší sám tyto standardní přístroje doma napodobit.

Pro tyto amatérské a výzkumné projekty jsou v naší zemi dostupné komunikační moduly s frekvencemi v pásmu pro průmyslové, vědecké a lékařské účely (industrial, scientific and medical - ISM). Dle [1] přístroje pracující na těchto frekvencích musí počítat s rušením svého signálu ostatními stanicemi a zároveň musí mít nízký výkon a krátký dosah pro omezení rušení jiných zařízení. Z frekvencí v tomto pásmu se u nás nejvíce používají 433,92 MHz; 868 MHz; 2,4 GHz a 5 GHz.

2.1 Pásmo 433,92 a 868 MHz

Tato pásma jsou povolena všeobecným oprávněním. Používat je tedy může kdokoli za dodržení podmínek nízkého vyzařovacího výkonu a tedy nízkého rušení ostatních přístrojů. Vzhledem ke složitosti vytvářet vlastní vysílací části se u nás prodávají hotové moduly připravené odvysílat přivedený signál. Tyto moduly jsou dobré pro vytvoření vlastního kódování signálu, nízkoenergetické aplikace jako jsou jednoduchá dálková ovládání, cílové části časomír ale také k výukovým účelům pro osvojení standardních přenosových protokolů. Používají převážně frekvenční modulaci signálu. Mají maximální přenosovou rychlost 50 kbps (kilobits per second). Dosah se pohybuje okolo 50 m se všesměrovou anténou nebo až 3 km se směrovou anténou.

Výhody těchto modulů jsou nízká spotřeba díky použití jednoduchých vysílacích obvodů bez řízení toku dat, z tohoto vyplývá i malá zastavěná plocha, obvykle jednoduchá komunikace s takovýmto modulem – co přivedeme na vstup, to modul odvysílá, protějšek na druhé straně přijme a objeví se na jeho výstupu. Tedy jednoduchá modulace na vysílaný signál. Avšak některé složitější moduly již obsahují i sériové rozhraní pro připojení například mikrokontroleru nebo výstupu z počítače. Tyto vyspělé moduly dále obsahují i funkce pro kontrolu toku dat a síly signálu – tyto funkce jsou však celkem primitivní a mají spíše informativní hodnotu pro uživatelský program, který s touto informací poté může nakládat.



Výhodou signálů o frekvencích 433 MHz a 868 MHz je jejich nižší náchylnost k utlumení zástavbou a terénními nerovnostmi. Jejich další výhodou může být i značně menší provoz na rozdíl od 2,4 GHz. Tím se snižuje rušení bezdrátové komunikace a výsledkem jsou menší nároky na sílu ochrany vysílaných dat. Je tak možné posílat data přímo nebo s jednoduchou ochranou typu parita atp.

Nevýhodou může být jedna z jeho předností (jednoduchost komunikace) – při rušení je signál téměř nemožné opravit, případně chyby objevit a zajistit opakování vysílaných dat. Takovéto funkce řízení toku dat by musel obsahovat řídicí mikrokontroler/počítač a tím by se zaplnila značná část jeho programu (ne-li celý). Další technologie tyto funkce řízení toku dat mají již vestavěné a z větší části vykonávané pomocí k tomu specializovaných součástek. Vzhledem k těmto nevýhodám jsem z řešení pomocí těchto modulů opustil.

2.2 IEEE802.15.4 a nadstavba ZigBee

Podle [2] byl standard IEEE802.15.4 uveden v roce 2003. Pracoval na 3 frekvenčních pásmech:

868,0 – 868,6 MHz s 1 komunikačním kanálem pro Evropu

902 – 928 MHz s 10 komunikačními kanály pro severní Ameriku (roku 2006 rozšířeno na 30)

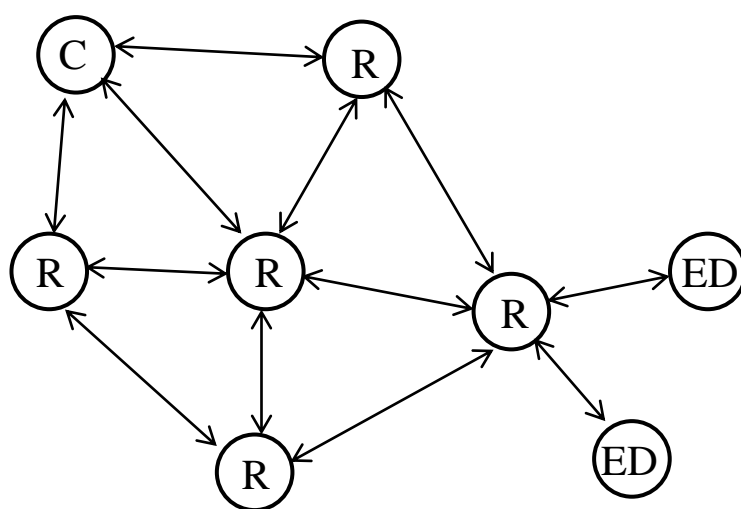
2400 – 2483,5 MHz s 16 komunikačními kanály pro globální použití

Rychlosti byly nastaveny na 20 a 40 kbps pro 868/915 MHz a 250 kbps pro 2,4 GHz. Revize roku 2006 zvýšila přenosovou rychlost 868/915 MHz na 100 a 250 kbps. Nyní je již standard rozšířen do mnoha frekvenčních pásem a rychlostí. Avšak stále nejpoužívanějším je globální pásmo 2,4 GHz, které využívá i nadstavba ZigBee.

Hlavním cílem ZigBee jsou nízko odběrové aplikace nevyžadující velké datové toky a velice jednoduché zabezpečené sítě (PAN) [3]. V těchto případech nabízí nižší cenu a spotřebu než konkurenční Bluetooth a Wi-Fi. Příklady použití jsou domácí „chytré“ spotřebiče, elektroinstalace, ale i celé „chytré domy“, dále víceúčelové dálkové ovladače, zdravotnické a fitness přístroje, mnohé počítačové periferie (myši, klávesnice, joysticky), multimediální technologie (dnes populární 3D obraz). ZigBee se však využívá i v průmyslu, kde se používají bezdrátové snímače a hlavně pro automatizace staveb.



Dosah je přibližně 75 m pro ZigBee a až 1500 m se ZigBee PRO. Výkon zařízení je obvykle 1 mW. S maximální rychlostí již výše zmiňovaných 250 kbps obsahuje 2,4 GHz pásmo 16 kanálů, každý o 5 MHz. Topologie sítě ZigBee zahrnuje několik typů zařízení. Hlavním je ZigBee Coordinator a v každé síti musí být alespoň jeden. Jeho úkolem je řídit tvar a zabezpečení sítě. Dále síť obsahuje ZigBee Routers, které kromě svého běžného programu (měření, řízení) navíc zvětšují dosah sítě a počet připojených zařízení. Třetím typem je ZigBee End Device, které již jen plní svůj program a dále síť nerozšiřuje. Mnoho výsledných modulů však integruje několik typů zařízení v jednom. Graficky znázorněný příklad sítě je na obrázku obr. 2.1.



Obr. 2.1: Topologie sítě ZigBee

Nejnovějším přírůstkem je ZigBee RF4CE, který sestává na rozdíl od klasického ZigBee z jiných stavebních prvků. Základními typy zařízení jsou cílový a ovládací uzel. Takováto topologie je zaměřena na multimediální domácí síť, kdy například televize, DVD přehrávač a CD rádio obsahují cílové uzly a mají každý svůj ovladač (ovládací uzel) a zároveň mají jeden ovladač nadřazený. Ten může kromě těchto zařízení ovládat ještě například osvětlení a další prvky související s multimédií.

Výhodami ZigBee jsou energetická nenáročnost, dobré zabezpečení přenosu (podporuje i AES-128), robustnost přenosu je zaručena díky DSSS (Direct Sequence Spread Spectrum) – rozptřeni signálu na širší frekvenční pásmo než je potřebné (vícenásobná redundance dat) [4]. Také obsahuje mnoho základních profilů a umožňuje vytvářet nové pro specifické aplikace. Avšak jeho použití pro amatérské a výzkumné aplikace je omezeno vyšší cenou a malým sortimentem u nás dostupných modulů. Dále má pro měřicí aplikace s více



snímači v jedné stanici nízkou přenosovou rychlost. Také je u nás zatím málo rozšířen. Z těchto důvodů jsem jeho aplikaci do měřicí úlohy nezvolil.

2.3 IEEE802.11 Wi-Fi

Tento bezdrátový standard určitě všichni známe a jeho i stručný popis by vydal na samostatnou práci. Takže se zaměřím jen na jeho možné použití pro průmysl a hlavně konkrétní laboratorní úlohu. Technologie Wi-Fi je určena jako alternativa metalických spojů v lokálních počítačových sítích. Má tedy velice dobré přenosové rychlosti – dnes teoreticky až 600 Mbps s 802.11n. Dosah je též velice slušný a pro průmyslové účely naprosto dostačující. Se všesměrovou anténou přibližně 250 m. Velkou výhodou Wi-Fi je opravdu silné zabezpečení dat dané používáním jako počítačové sítě. Robustnost přenosu je podobná jako u ZigBee tedy pomocí DSSS, ale navíc ještě používá FHSS (Frequency Hopping Spread Spectrum) při kterém se přeskakuje mezi jednotlivými kanály – přibližně 2,5 krát za sekundu [4].

Průmyslové využití nachází Wi-Fi hlavně jako spojení sítí senzorů a řídicích terminálů s firemním intranetem nebo jako pro připojení k internetu. Například pro spojení serveru v IT části výrobní haly s jednotlivými obráběcími stroji – takovéto spojení vyžaduje vysokou míru zabezpečení a také dostatečnou rychlost dat. Pro přímé připojení ke skupinkám senzorů je tato technologie značně neefektivní vzhledem k její vysoké spotřebě, složitosti a ceně. I přes dobrou dostupnost modulů u nás jsem se vzhledem k ceně, složitosti a nepříznivým parametrům jako spotřeba a velikost modulů rozhodl tuto technologii při vývoji bezdrátového odměřování na laboratorní úloze letadlo nevyužít.

2.4 IEEE802.15.1 Bluetooth

Jedná se o asi nejznámější standard pro osobní sítě (PAN) a bezdrátové příslušenství. Standard byl vytvořen roku 1994 společností Ericsson. Ale až v roce 1998 se několik společností dohodlo na společné spolupráci a propojení svých produktů právě pomocí Bluetooth. Tímto byla založena Bluetooth Special Interest Group (SIG) – organizace pro udržování a rozvoj této bezdrátové technologie [5]. Právě jméno „Bluetooth“ standard dostal až od této organizace. Jméno pochází už z 10. století, kdy dánský král Harold Bluetooth sjednotil kmeny na území dnešního Norska, Švédska a Dánska. Stejně tak i bezdrátová



technologie Bluetooth sjednocuje protokoly pro odvětví počítačů, mobilních přístrojů a průmyslové automatizace.

Hlavním cílem vytvoření IEEE802.15.1 bylo nahrazení metalické sériové linky RS232 pro propojení mezi mobilními přístroji a například tiskárnami nebo dalšími zařízeními. Díky tomu je jeho nejzákladnějším profilem simulace sériové linky (SPP), kterou bude mé zapojení využívat. Bluetooth pracuje v globálním ISM pásmu na frekvencích 2,4 – 2,485 GHz. Využívá jak DSSS tak FHSS (vysvětlené výše) pro zajištění robustnosti přenosu. Dále díky AFH (adaptive frequency hopping) zvyšuje svoji odolnost proti rušení dalšími 2,4GHz technologiemi. AFH funguje na principu detekce ostatních zařízení a tím se dokáže vyhnout frekvencím, které používají. Pro toto adaptivní přeskakování na různé frekvence má ve svém pásmu 79 možných frekvencí s intervalem 1 MHz. Díky tomuto je Bluetooth vysoce odolný proti rušení. Dosah (a tím i výkon) modulů se dělí na tři třídy:

Class 3 – dosah do cca 1 m

Class 2 – dosah do 10 m (primárně mobilní telefony a příslušenství)

Class 1 – dosah až 100 m (primárně průmyslové aplikace)

Zmíněné dosahy jsou se všesměrovými anténami. Třetí třída s nejkratším dosahem se díky nízké spotřebě hodí pro miniaturní bateriově napájené přístroje, jako jsou nejjednodušší bezdrátová sluchátka a handsfree. Její výkon se pohybuje okolo 1 mW. Druhá třída s typickým výkonem 2,5 mW je nejvíce zastoupena v mobilních přístrojích jako jsou telefony, PDA, tablety a některých noteboocích. Nejvýkonnější třetí třída není svým odběrem příliš určena pro malé přístroje, avšak díky pokročilým úsporným technologiím by se i pro tyto aplikace mohla použít. Samozřejmě by ovšem byly jen krátké datové toky s pauzami, nikoliv jeden nepřetržitý. Výkon třetí třídy je maximálně 100 mW.

Technologie Bluetooth se od Wi-Fi liší hlavně v aktivní účasti i ve vyšších vrstvách než jen linkové. Jeho účast sahá až do aplikační vrstvy. Tím je dána nutnost používat pro různá připojená zařízení různé profily. Nejznámější profily jsou například pro simulaci sériové linky SPP, pro přenos souborů profily OBEX, profily pro ovládání (klávesnice, myš) HID, osobní síť PAN, nebo například pro bezdrátové modemy DUN. Jak už bylo řečeno, pro každou funkci zařízení je nutné mít speciální profil, tím je výčet profilů opravdu veliký. Přenosová rychlost Bluetooth dosahuje 3 Mbps v nejrozšířenější verzi 2.0. V nejnovější verzi 4.0 je to teoreticky až 24 Mbps. Propustnost dat je však nižší – u verze 2.0 je to 1,4 Mbps.



Nevýhodami může být právě tato nutnost specializovaných profilů. Nižší datový tok u standartní verze by mohl být překážkou u zařízení vysílajícího video případně vícekanálové audio s vyšším datovým tokem.

Výhody technologie Bluetooth:

- jeho masivní celosvětové použití, díky kterému je přidávání nových systémů do stávajících jednodušší.
- universálnost a díky hotovým profilům též jednoduchý vývoj software.
- vysoká odolnost proti rušení a tím i robustnost přenosu dat.
- dostačující rychlost přenosu pro průmyslové měření a regulaci.
- dobrá dostupnost a cena modulů u nás.

Výše uvedené z něj činí favorita pro aplikace výzkumu a vývoje. S ohledem na tyto výhody jsem se rozhodl použít pro svou práci právě Bluetooth moduly nahrazující sériovou linku.

Nakonec je zde ještě uvedena stručná tabulka (tab. 2.1) porovnávající uvedené bezdrátové komunikace.

Tab. 2.1: Srovnání bezdrátových komunikací

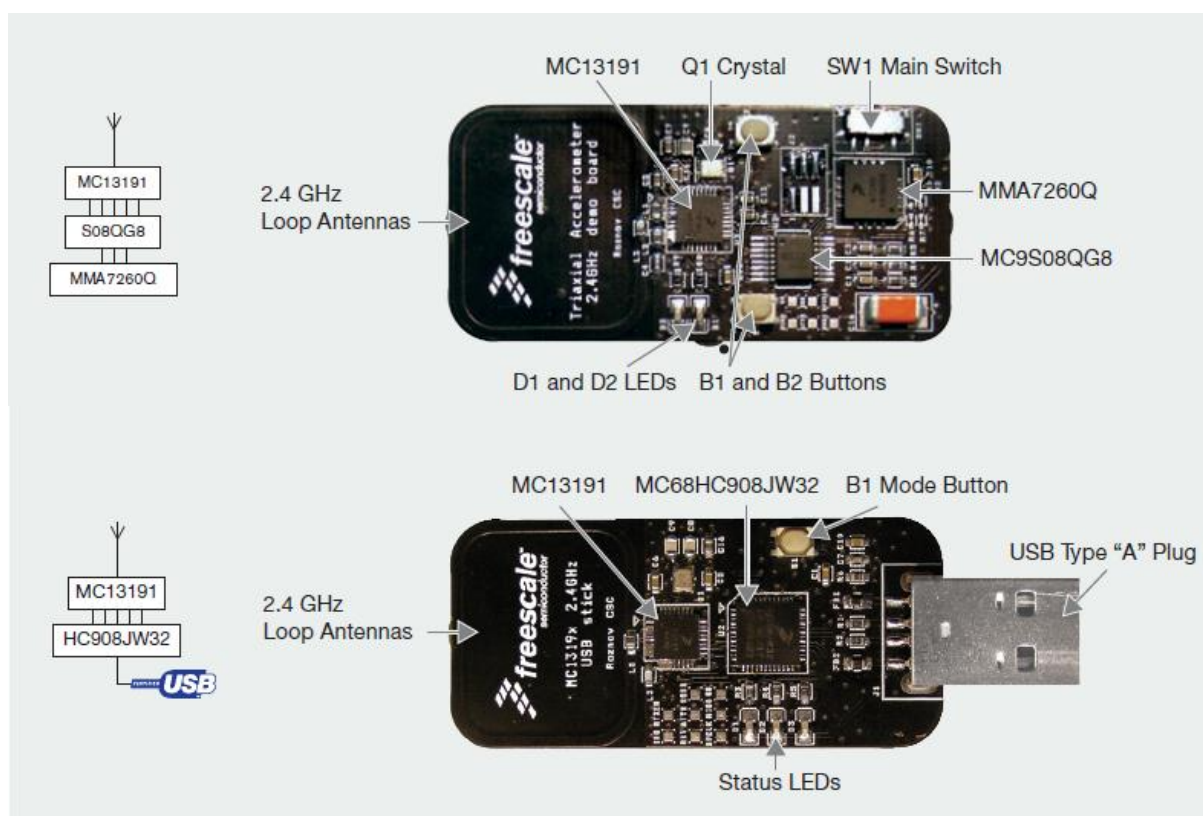
Technologie	Volné frekvence	IEEE802.15.4 ZigBee	IEEE802.15.1 Bluetooth	IEEE802.11 Wi-Fi
Frekvence	433/868 MHz	2,4 GHz	2,4 GHz	2,4/5 GHz
Přenosová rychlost	50 kbps	250 kbps	3 Mbps	600 Mbps
Dosah	~50 m	~75 m	~100 m	~250 m



3 Možné návrhy řešení bezdrátového odměřování

3.1 Freescale ZSTAR

Prvním možným řešením je využití hotového modulu bezdrátového odměřování. Tento konkrétní je od firmy Freescale. ZSTAR je kompletní řešení tříosého akcelerometru, řídicího mikropočítače a ZigBee transceiveru pro přenos [6]. Jedná se však o tzv. demo modul, který je přímo k počítači připojen pomocí USB. Tím vyžaduje vlastní obslužný software, který je jen stěží kompatibilní s prostředím Matlab a jeho měřicí kartou. Dále neumožňuje připojení dalšího signálu pro přenos ani zpětné signály. Výhodou ZSTAR jsou jeho malé rozměry a díky USB jednoduché připojení k počítači. Vzhledem k uvedeným vlastnostem ZSTAR není jeho použití pro úlohu vhodné. Avšak osazený akcelerometr Freescale MMA7260Q je zajímavá součástka. Bude jí věnována jedna z kapitol popisu osazených senzorů. Na obr. 3.1 jsou moduly ZSTAR.



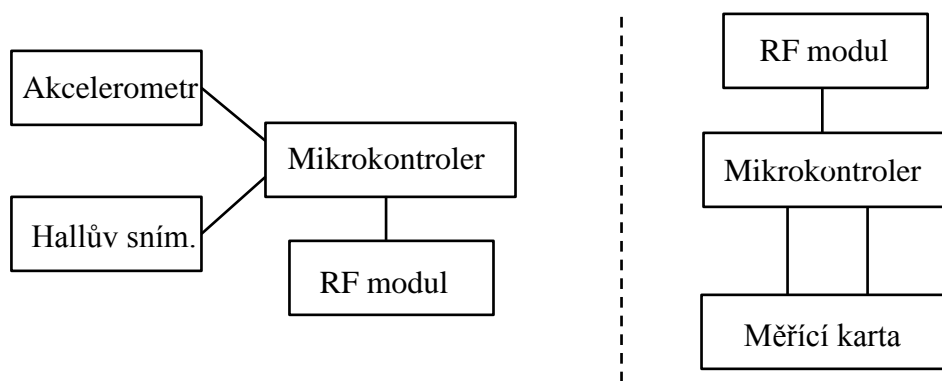
Obr. 3.1: Moduly ZSTAR (www.freescale.com)



3.2 Použití modulů 433/868 MHz

Další možností je použití modulů pro volné frekvence. Pro jejich ovládání by bylo možné použít mikrokontroler PIC. Výhodou tohoto spojení je volitelná modulace a neomezená délka dat. Tedy v případě použití 16 bitového mikrokontroleru PIC by komunikace mohla být rovnou 16 bitová. K mikrokontroleru by se na měřicí straně připojily zvolené snímače zrychlení a již vestavěný snímač polohy. Vyhodnocující strana by obsahovala opět modul 433/868 MHz, mikrokontroler PIC pro demodulaci dat a převod signálu na analogový jako výstup signálu do měřicí karty počítače. Například pomocí PWM a integračního článku.

Na obr. 3.2 je blokové schéma takového řešení, kde měřicí strana je vlevo a vyhodnocující vpravo.



Obr. 3.2: Blokové schéma s 433/868 MHz moduly

Výhodami takového zapojení by byla prostorová nenáročnost a jednoduchá komunikace bezdrátových modulů díky volitelnému protokolu. Tento přenosový protokol by pak měl obsahovat alespoň základní funkce jako definovaný „start bit“ a „stop bit“ pevně danou délku dat. Pokud by měl mikrokontroler dostatek paměti programu, tak by se mohl naprogramovat například některý ze samo-opravných kódů. Avšak poté by mohla značně klesnout propustnost užitečných dat. Nevýhodou tohoto řešení je už v základu nízká přenosová rychlost a také jak již bylo zmíněno výše, moduly neobsahují žádnou vlastní korekci přijatých dat. Vystaly by tedy velké nároky na program v mikrokontroleru.



3.3 Wi-Fi měřicí úloha

Velice zajímavým řešením úlohy by bylo Wi-Fi spojení. Měřicí modul na letadle by obsahoval snímače, mikrokontroler a tento bezdrátový vysílací modul. Po změření hodnot na výstupech snímačů a jejich vyhodnocení mikrokontrolerem, by tento odeslal data do Wi-Fi modulu. K tomuto by poté mohl být připojen jakýkoli počítač obsahující kartu nebo modul pro bezdrátovou komunikaci. Také by bylo možné takto získaná data posílat do internetu. Výhodou takového řešení by byla možná absence osob při provozování pokusů a tedy zajištěná 100% bezpečnost. Možností a výhod tohoto uspořádání by jistě mohli využít studenti dálkového studia, kteří se do učeben a na laboratorní úlohy nedostávají tak často jako studenti prezenčního studia.

Zásadním problémem takto otevřené Wi-Fi úlohy je její řízení. Stávající řešení podmiňuje účast v učebně a navíc jen na jednom počítači. Tudíž by bylo nutné buď tento počítač též připojit k síti a zpřístupnit ho z internetu, nebo přepracovat řídicí obvody úlohy pro Wi-Fi ovládání. Dále vyvstává problém zpracování dat v cílovém počítači připojeném k úloze. Ten by totiž musel obsahovat konverzní program, který data přijatá ze sítě převede do programu Matlab. A zároveň řídicí povely z programu Matlab převede do formátu, který by byl poté odeslán do sítě. Nároky na mikrokontroler v případě připojení do internetu by byly značné vzhledem k nutnosti webového rozhraní a věcí s tím souvisejících. Pro lokální měření – tedy takové, které je v dosahu vysílání Wi-Fi, by se modul mohl omezit na například simulaci sériového portu [7], a tudíž by takovéto měření zvládl i obyčejný mikrokontroler. Stále však by byla nutnost převést data z tohoto simulovaného sériového portu v počítači do programu Matlab.

3.4 Vývojové ZigBee moduly RC220x

Tyto moduly vyrábí firma Radiocrafts [8]. V jednom stíněném SMD pouzdru integrují ZigBee transceiver a mikrokontroler AVR ATmega [9]. Díky technologii ZigBee má modul nízkou spotřebu a velký dosah. Osazený mikrokontroler ATmega disponuje až 128kB flash paměti, až 4kB RAM, 32 porty, 8kanálovým 10bitovým AD převodníkem a rozhraními UART, SPI a JTAG. Modul s těmito parametry by byl téměř ideální pro bezdrátové odměřování. Bohužel absence jakýchkoli zkušeností s programováním mikrokontrolerů firmy Atmel by vývoj výrazně prodloužilo. Je zde též požadavek na výměnný mikrokontroler pro případ opravy, což RC220x nenabízejí. Integrace vysílacího modulu a ovládacího jednočipu



s sebou přináší též nutnost umístění do místa snímání kvůli absenci dlouhých vodičů ke snímačům, jež mohou mít analogový výstup náchylný k rušení. Snímač zrychlení musí být umístěn tak, aby nežádoucí pohyby měřeného objektu neovlivňovali měření. Například pootočení měřeného osového systému od skutečného. Takovéto místo však může být v rozporu s vysíláním modulu – stínění, rušení a podobné vlivy by značně zkrátily dosah vysílání. V tomto ohledu je výhodnější varianta odděleného jednočipu od vysílacího modulu. Jejich komunikace je již digitální a tedy odolnější proti rušení. Takto by mohl být vysílací modul umístěn na pro vysílání výhodném místě. Také jeho špatná dostupnost u nás nehovoří v jeho prospěch. Připojení Zigbee je sice úsporné, ale nemá velkou přenosovou rychlost. Též robustnost přenosu je u Bluetooth vyšší. Výsledné zapojení by bylo obdobné jako u předchozí varianty s moduly na volných frekvencích, jen řídicí mikrokontroler a vysílací modul by byly sloučeny do jednoho. Zejména z důvodu špatné dostupnosti nebylo toto řešení pro úlohu vybráno.



4 Popis vybraného řešení – modul pro letadlo

Z výše uvedených alternativ řešení je zřejmé, že hledané řešení by mělo využívat Bluetooth komunikační modul se sériovou komunikací. Dále zapojení musí obsahovat výkonný mikrokontroler v patici, čímž půjde snadno vyměnit. Též je nutné osazení inerciálních snímačů – akcelerometru a gyroskopu. Transceiver musí být oddělen od mikrokontroleru se snímači pro jeho správné umístění vyhovující vysílanému signálu. Na tento požadavek se dá nahlížet i z pohledu snímačů, kdy mikrokontroler by měl být k těmto co nejbližší za účelem zkrácení možné cesty analogového signálu.

Všechny tyto podmínky splňuje mé řešení obsahující mikrokontroler od firmy Microchip, komunikační modul Bluetooth společnosti connectBlue, akcelerometr Freescale a gyroskop InvenSense.

4.1 Mikrokontroler

Jedná se o řadu PIC a model 24F04KA201 [10]. Kromě jiných se prodává i v klasickém plastovém DIP pouzdře. Koncové označení „KA201“ znamená, že tento konkrétní model disponuje 20 piny a jde o mikrokontroler pro všeobecné použití, stejný čip se ještě prodává ve verzi „KA200“, která má však jen 14 vývodů. Méně vývodů by se mohlo zdát jako lepší pro úlohu s požadavkem na co nejmenší zastavěnou plochu, avšak méně pinů znamená i více funkcí na jednom a tím vyvstává možná kolize několika funkcí neslučitelných pro vyvíjené zapojení. Také by samozřejmě nemusel tento počet portů pro úlohu postačovat. Dvojice čísel „04“ označuje 4kB paměti programu a první dvojčíslí s písmenem „24F“ rodinu 16-ti bitových jednočipů s pamětí typu Flash.

Rozdíl mezi konstruktéry hojně využívanými rodinami PIC12, 16 a 18 a řadou 24F je 16 bitová architektura, která usnadňuje operace s většími čísly. Šetří tak řádky programu a instrukční cykly. Dále jsou zde změny v počtu pracovních registrů, kdy 8 bitové mají jen jeden, kdežto PIC24 jich obsahuje 16. Toto má též značný dopad na rychlost zpracování dat programem. Asi posledním opravdu zásadním rozdílem, též velmi ovlivňujícím rychlost a výkon mikrokontroleru, je počet period oscilátoru na vykonání jedné instrukce. Jednočipy rodiny PIC18 (funkcemi nejbližší zvolenému PIC24F) potřebují čtyři periody oscilátoru na vykonání jedné instrukce, ale vybraný jednočip jen dvě. Přičemž oba mají podobné maximální frekvence oscilátorů. Další rozdíly jsou také důležité, ale ne pro všechny aplikace použitelné. Například větší výběr délky datových formátů (byte, word a double word oproti



jen bytu u PIC18), možnosti použití hardware akcelerovaného dělení, větších možností násobiček, několik stupňů důležitosti přerušení a podobně, viz [11].

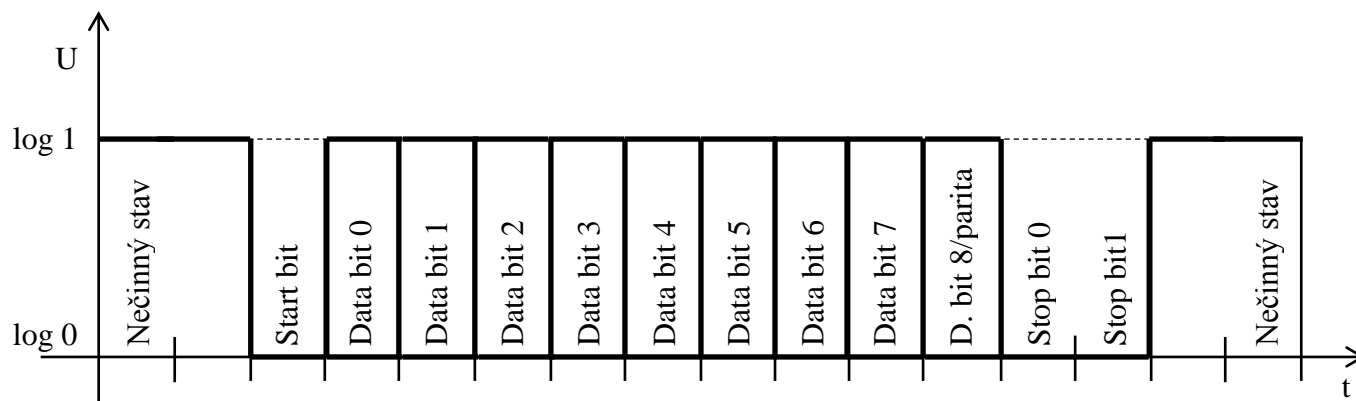
Mikrokontroler je určen pro napájení napětím od 1,8 V do 3,6 V. Připojený oscilátor může mít frekvenci až 32 MHz. Programování probíhá pomocí ICSP (In-Circuit Serial Programming – programování jednočipu v hotovém zapojení). U tohoto konkrétního typu jsou k dispozici dvě sady datových vstupů, aby bylo možné tyto vývody nezačleňovat do zapojení. Dále bude uvedeno několik důležitých funkcí mikrokontroleru. Jejich popis by se mohl zdát až příliš podrobný a opakující všeobecně známá fakta, ale je to nutné pro pochopení souvislostí a problémů vzniklých během vývoje a jejich řešení.

Pro funkci měřicího obvodu se u tohoto jednočipu bude využívat v první řadě analogovo-digitální převodník (A/D převodník). Tento je řešen pomocí metody postupné aproximace. Je to 10bitový převodník s rychlostí až 500 kps (kilosample per second – tisíc vzorků za sekundu). Je tedy velmi rychlý – nejkratší doba jednoho cyklu je 75 ns, přičemž volitelný počet je pro vzorkování a podržení hodnoty a 12 jich je potřeba na samotný převod [12]. Jedná se o samostatný modul, program tedy řídí jen parametry a začátek sekvence vzorkování a následného převodu. Převodník poté již pracuje sám a na konci zvolených operací výsledky převodů zapíše do svého zásobníku a nastaví příznak přerušení.

Další použitou funkcí jednočipu je modul UART (universal asynchronous receiver/transmitter). Jedná se o velmi jednoduchou komunikaci, která je téměř totožná se sériovou linkou RS232. Jediným rozdílem jsou napěťové úrovně, kdy RS232 používá symetrické napájení a tedy i signál – nejčastěji $\pm 12\text{V}$. UART má signál od 0 V do 3,3 V (kladné napájecí napětí). Jak je již z názvu patrné, jedná se o komunikaci asynchronní, tedy musí být obě strany domluveny na rychlosti přenosu a podobě datového paketu. Nečinná linka UART je v logické „1“, tedy téměř plné napájecí napětí. Začátkem posílaných dat je tzv. „start bit“, který je definován jako nulový bit. Ten následuje 8 nebo 9 bitů dat (podle zvoleného nastavení). Při použití 8bitových dat je obvykle 9. bit použit pro paritu. Avšak nemusí být vůbec nastaven a nebude tedy odvíšlán. Poté následuje jeden nebo dva „stop bity“ – též nulové bity, počet je opět závislý na nastavení a na domluvě komunikujících stran. Modul UART je stejně jako analogovo-digitální převodník také automaticky řízen svými vnitřními obvody. Program tedy určuje jen nastavení přenosu a poté zapisuje nebo vyčítá příchozí data. Tyto se zapisují ve formě bytu (8 bitů dat) nebo wordu (9 bitů dat) do zásobníku modulu a program se tak nemusí zabývat přímo uskutečněním přenosu. Přesto, že



se jedná o jednoduchou komunikaci, její spojitost s linkou RS232 dovoluje pouze relativně nízké přenosové rychlosti. Ty se pohybují do přibližně 3 Mbps, avšak vzhledem k použití sériových linek se u nastavení jejich ovladačů objevují hodnoty nižší. Na obr. 4.1 je znázorněn datový paket UART.



Obr. 4.1: Datový paket komunikace UART

Poslední z funkcí, které by tu bylo vhodné popsat, jsou integrované pull-up rezistory. Zapojení klasických (externích) pull-up rezistorů je nutné například pro vývody s otevřeným kolektorem, kdy se přes tyto dostává na signálový vodič kladné napětí. Dále se používají pro ovládací prvky jako tlačítka a klávesnice, které nemají aktivní výstup a fungují pouze jako připojení k jedné z polarit napájecího napětí. Zde je možnost i opačného způsobu, tedy pull-down rezistoru, který naopak klidový stav napětí drží blízký nule. A právě v případě připojení ovládacích prvků k mikrokontroleru, může vyvstat problém s umístěním rezistorů. Proto je jejich integrace přímo do jednočipu velmi vítanou funkcí. Řešeny jsou pomocí proudových zdrojů s proudem o hodnotě přibližně 250 μA [10]. Tato funkcionalita je však dostupná jen pro čistě digitální vstupy/výstupy. Porty s některou z analogových funkcí (komparátor, analogovo-digitální převodník) tuto možnost nemají.

4.2 Bluetooth transreciever

Pro komunikaci je použit Bluetooth modul od švédské společnosti connectBlue. Jeho označení je OEMSPA310i [13]. Jedná se o nejjednodušší Bluetooth 2.0 modul, který firma vyrábí. U nás výrobky connectBlue distribuuje internetový obchod special.cz. Označení „i“ na konci znamená, že je na desce modulu už integrována všesměrová anténa. Jeho funkce se dá měnit v závislosti na nahraném firmware. Ten základní je pro simulaci sériové linky RS232 tedy na principu profilu SPP, ale rozhraní je typu UART (jen jiné úrovně napětí). Nahráním jiného firmware se z modulu může stát vícekanálové 1/0 dálkové ovládání viz [14]. Nebo

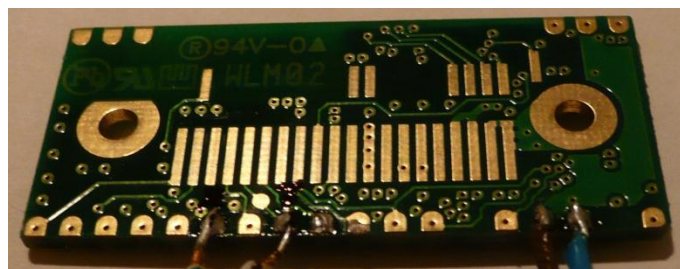


například Bluetooth opakovač pro zvýšení dosahu připojených zařízení. Značnou výhodou použitého modulu je jeho nenáročnost na externí součástky a program. Po zakoupení je tedy ihned připraven k práci, stačí jen připojit stejnosměrné napájecí napětí 3 – 6 V. Toto napětí ani nemusí být vyhlazeno díky lineárnímu stabilizátoru integrovanému v modulu. S výkonem 2 mW se řadí do 2. třídy zařízení Bluetooth. Jeho dosah by tedy měl být přibližně 10 m, což je pro tuto laboratorní úlohu naprosto dostačující. Spotřeba modulu není vysoká, avšak s jinými řešeními, případně dalšími generacemi technologie Bluetooth se nemůže rovnat. Pro úlohu však toto není důležité vzhledem k pevnému napájení ze síťového zdroje. Špičková spotřeba při vysílání spárovaného modulu dosahuje maximálně 58 mA. Nejvyšší spotřebu má modul při hledání dalších Bluetooth zařízení – tehdy může vystoupat až na 70 mA.

S firmware pro nahrazení metalické sériové linky modul disponuje vstupem pro příjem dat (RxD), výstupem odesílání dat (TxD), vývody pro kontrolou toku dat a informačními výstupy pro LED signalizaci. Na těchto signalizačních vývodech je též přidružena funkce resetování do původního stavu. I přes možnost napájení až 6 V stejnosměrného napětí, operují všechny datové a informační vstupy a výstupy na 3V logice. Toto je velmi výhodné vzhledem k použití s mikrokontrolerem též operujícím na této napěťové úrovni. Na obr. 4.2 je pohled na Bluetooth modul zepředu, obr. 4.3 ukazuje zadní stranu s vývody.



Obr. 4.2: Přední strana Bluetooth modulu



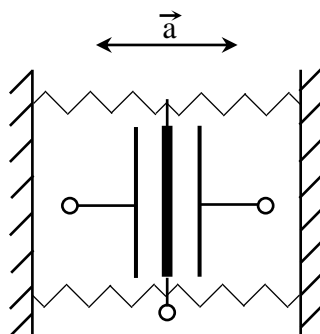
Obr. 4.3: Zadní strana Bluetooth modulu

4.3 Akcelerometr

Snímač zrychlení je použit od firmy Freescale, jedná se o model MMA7260Q [15]. Prodává se jen v QFN pouzdře s 16 vývody. Tento MEMS (micro-electro-mechanical system) akcelerometr pracuje ve všech třech osách. Jeho rozsah je nastavitelný a má čtyři úrovně: $\pm 1,5$ g; ± 2 g; ± 4 g a ± 6 g. Citlivost při rozsahu $\pm 1,5$ g je 800 mV/g. Tímto je předurčen pro velice universální použití. Stejnosměrné napájecí napětí může být v rozmezí 2,2 – 3,6 V, zde je opět výhoda tohoto rozsahu vzhledem k ostatním použitým součástkám. Jeho spotřeba se pohybuje kolem 500 μ A. Princip snímání je na bázi změny kapacit proměnného dvojitého



kondenzátoru v závislosti na zrychlení působícím na střední elektrodu. Na obr. 4.4 je znázorněn zjednodušený systém snímání.



Obr. 4.4: Soustava snímání zrychlení

Výstupem akcelerometru je analogový signál, který má klidovou hodnotu na polovině napájecího napětí, aby bylo možno určit jakého směru měřené zrychlení je. Výhodou tohoto snímače je integrovaný filtr typu dolní propust a integrovaná teplotní kompenzace.

4.4 Gyroskop

Osazený elektronický gyroskop pochází od společnosti InvenSense. Typové označení je ISZ-500 [16]. Označení gyroskop může pro někoho být zavádějící, protože klasické mechanické gyroskopy po roztočení, buď zůstávají stále v poloze, v jaké byly roztočeny, nebo ukazují na sever (díky rotaci Země). Tento gyroskop však měří úhlové zrychlení. Stejně jako akcelerometr je to součástka typu MEMS a pracuje na principu velice podobném. V gyroskopu je pružně uložena kmitající hmota (frekvence kmitání je přibližně 30 kHz), která se při otáčení díky působení Coriolisovy síly nerovnoměrně natáčí vůči svému závěsu. A tyto odchylky v poloze měří stejná kapacitní čidla, jako jsou v akcelerometru. Skutečný tvar celého „soustrojí“ je u každého výrobce jiný. ISZ-500 je jednoosý elektronický gyroskop s rozsahem ± 500 °/s. Obsahuje také integrovaný zesilovač, který zvyšuje citlivost z 2 mV/°/s na 9,1 mV/°/s a tím je dán druhý výstupní rozsah ± 110 °/s. Tento rozsah je již mnohem vhodnější pro pomalý odklon letadla od osy rotace. Jeho měřená osa je kolmá na plochu pouzdra a má dle pravidla pravé ruky (palec s kladným směrem osy – prsty ukazují kladný směr rotace okolo této osy) levotočivý kladný smysl. Prodává se jen v pouzdře QFN s 28 vývody. Napájení snímače je stejnosměrných 2,7 – 3,3 V, tímto se opět shoduje s ostatními součástkami v modulu pro letadlo. Pro rozkmitání mechanické struktury se však používá napětí vyššího a je tedy nutné k součástce připojit externí kondenzátor. Tento kondenzátor je připojen do obvodu nábojové pumpy, která vytváří napětí přes 12V. Další externí součástky



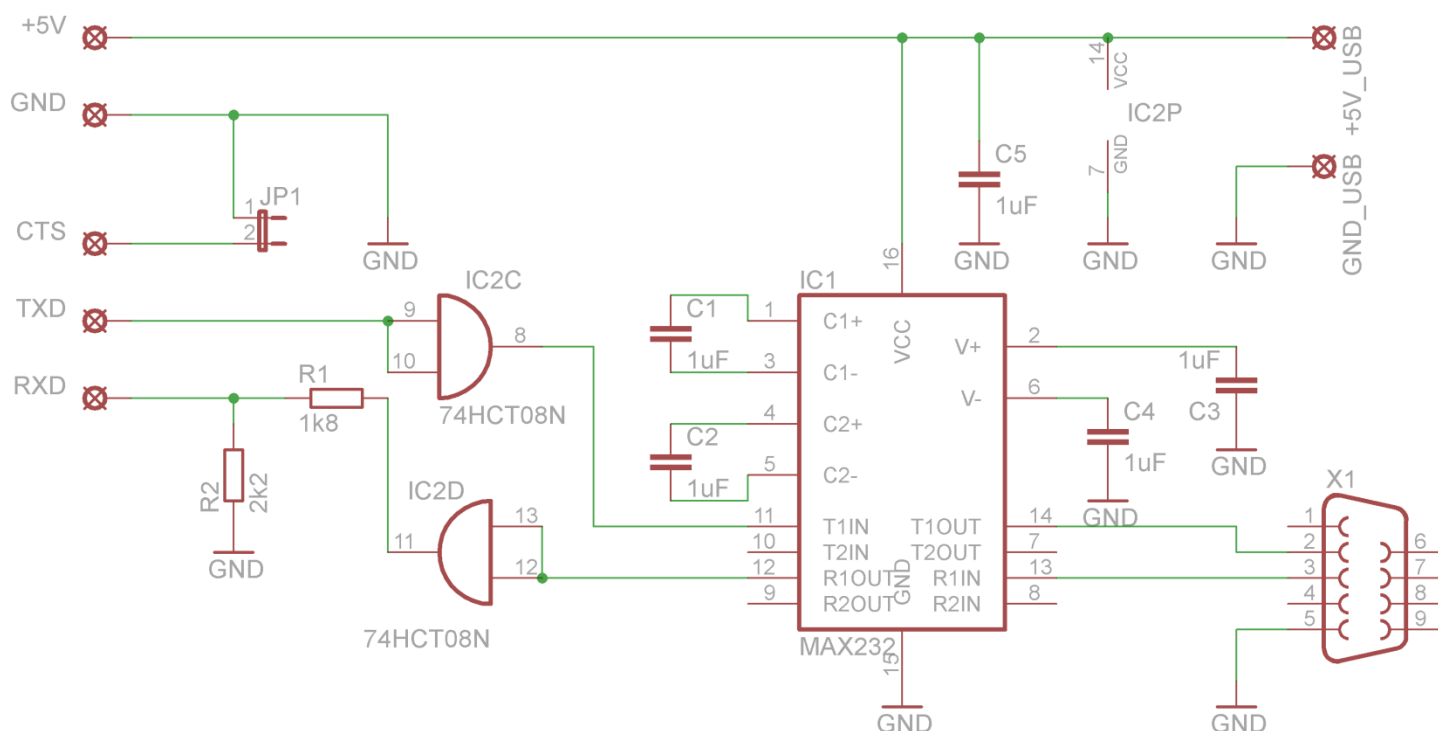
nutné pro funkci jsou rezistor a kondenzátor na výstupu, tvořící dolní propust. Vzhledem ke kapacitě kondenzátoru tato nemůže být integrována do pouzdra součástky. Po průchodu dolní propustí se signál vrací do součástky, aby byl zesílen a získal požadovaný rozsah $\pm 110^\circ/\text{s}$.

Výstup snímače je analogový, avšak jeho klidová hodnota není vztažena k polovině napájecího napětí, ale k integrované referenci o hodnotě 1,35 V. Tento elektronický gyroskop dále umožňuje nulování výstupu. Toho se využívá pro nastavení nuly v klidu, nebo pokud se pohybuje se stálým úhlovým zrychlením, které je známé. Tímto lze zvýšit rozsah – při nulování se struktura zastaví, znovu rozeběhne a tak vznikne posun celého rozsahu střední hodnotou právě na toto známé úhlové zrychlení.

4.5 Popis vývoje a funkce zapojení modulu pro letadlo

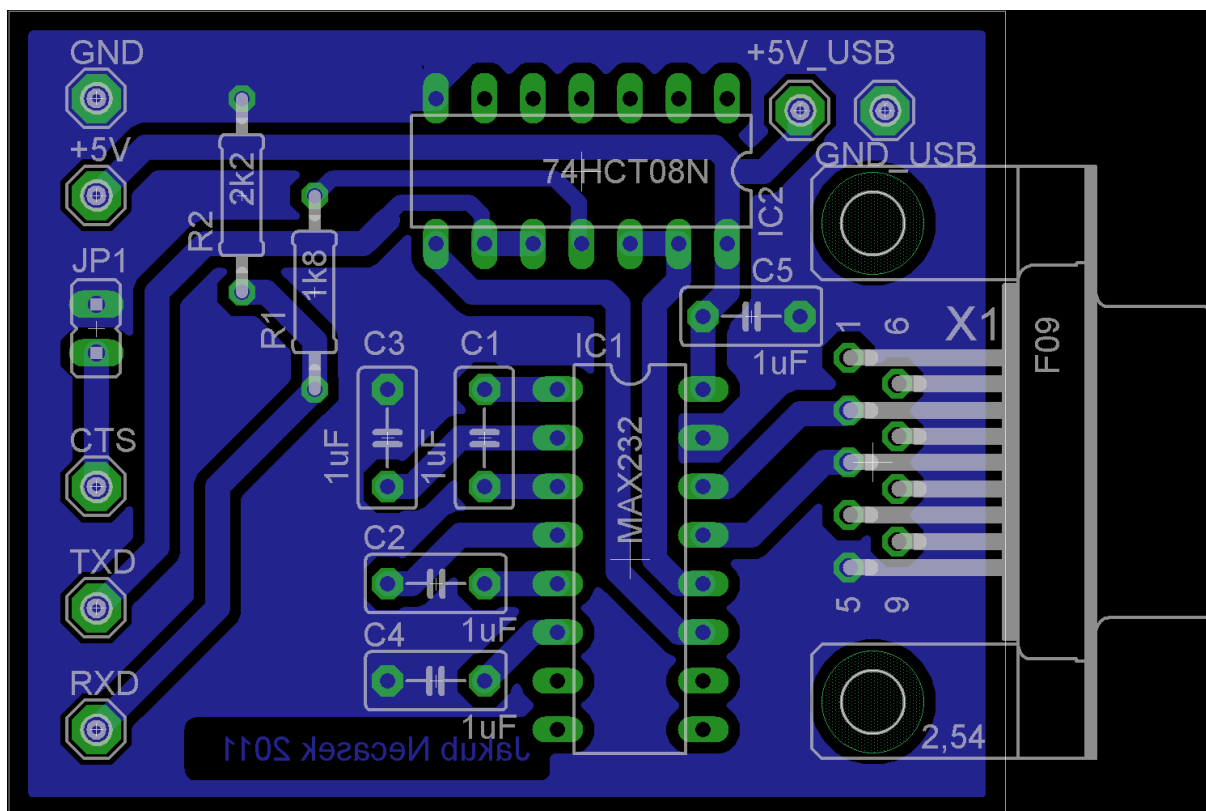
Zde bych rád popsal postup vývoje modulu osazeném v letadle. Prvním úkolem bylo nastavit Bluetooth moduly na vzájemnou komunikaci – spárovat je. V základním nastavení nejsou tyto moduly konfigurovatelné přímo pomocí Bluetooth z počítače, ale tato funkce se musí nejdříve povolit. Abych toto mohl provést, musel jsem zkonstruovat převodník ze sériové linky počítače RS232 na UART komunikaci Bluetooth modulu. Schéma zapojení takového převodníku je uvedeno i na stránkách distributora modulu [17]. Bohužel použité součástky nejsou u nás dobře k sehnání. Jedná se především o RS232 \leftrightarrow TTL převodník s napájecím napětím 3,3 V. Použil jsem proto 5V převodník, který je u nás běžně k dostání a to od mnoha výrobců. Já si vybral MAX232N v plastovém pouzdře DIP16 [18], který jsem vložil do patice a může se tak v případě poruchy vyměnit za nový. Díky 5V napájení se tedy dá napájet i z USB portu počítače a nepotřebuje externí zdroj. Tímto ale vzniká problém s nekompatibilitou úrovní převodníku (0 – 5 V) a Bluetooth modulu (0 – 3 V). V případě posílání signálu do modulu se řeší jednoduchým odporovým děličem. Avšak signál z modulu má příliš malé napětí pro převodník. Vložil sem tedy mezi převodník a modul jednoduchá hradla AND se spojenými vývody. Obvod 74HCT08D je velmi rychlý a je uzpůsoben TTL úrovní signálu, je tedy napájen také 5 V. Obvod se prodává v plastovém pouzdře DIP14 a je uložen v patici. Použitím hradla se výstup z převodníku posílil a je již zcela bezpečně použit odporový dělič pro přizpůsobení napětí. Signál z modulu hradlo bez obtíží rozezná díky spodní hranici napětí pro logickou „1“ rovnému 2 V. Takto je signál zesílen na 0 – 5 V pro převodník. Schéma zapojení celého převodníku je na obr. 4.5. Všechna schémata jsem kreslil

v programu EAGLE ve verzi 5.10. Licenci pro kreslení desek plošných spojů větších než dovoluje verze volně ke stažení, mi poskytla univerzita.

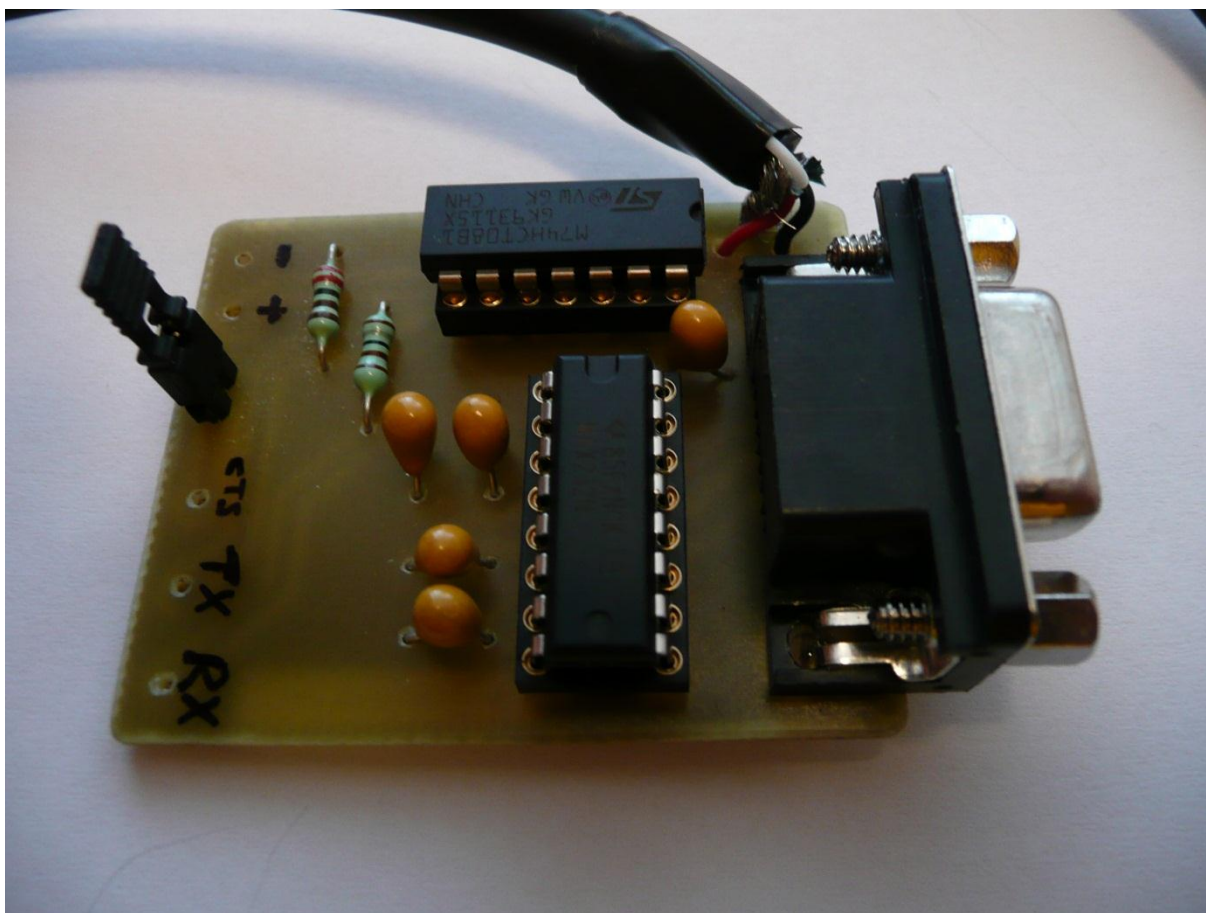


Obr. 4.5: Schéma převodníku RS232 – UART

Kondenzátory C1 až C4 jsou v integrovaném obvodu zapojeny do nábojových pump pro vytvoření napětí +12V a -12V. Volitelná propojka JP1 slouží k připojení CTS (clear to send – volno k poslání dat) vývodu Bluetooth modulu. Tento se při prvotním připojení k počítači musí uzemnit. Pro připojení k počítači je osazen konektor s 9 piny typický pro sériový port. Osazena je „samice“ aby bylo možno převodník zasunout přímo do konektoru počítače. Jako napájení se používá napevno připájený USB kabel se zapojenými jen napájecími a zemnicími vodiči. Na druhém konci napájecího kabelu je klasický USB A konektor. Na obr. 4.6 je deska plošných spojů i s vyobrazeným osazením součástkami pro toto schéma. Na obr. 4.7 poté hotový převodník.



Obr. 4.6: Deska plošných spojů i se součástkami pro převodník



Obr. 4.7: Hotový převodník



Po zhotovení programovací interface, jsem na počítač nainstaloval program pro nastavení modulů. Tento program je od distributora ke stažení zdarma po zakoupení modulů a nese název „Serial Port Adapter Toolbox“. Při spuštění program kontroluje zapojené sériové porty. Je tedy nutné převodník s Bluetooth modulem připojit před jeho spuštěním. Transciever se připojí k převodníku napájecími a datovými vodiči, zasune se jumper pro uzemnění CTS vývodu a nakonec se zapojí interface do počítače. Po spuštění programu jsem v úvodní obrazovce zvolil nastavení pro použité moduly – tedy „Bluetooth SPA“. V programu se stiskne tlačítko „Connect“ a tím se otevře podokno pro zvolení komunikačních parametrů. Zde se vybere sériový (COM) port, ke kterému se převodník připojí. Z výroby je modul nastaven na komunikační rychlost 57600 baudů za sekundu, 8 datových bitů, žádná parita, jeden stop bit, a protože je připojen vývod CTS na zem tak se zvolí řízení toku CTS/RTS. Nyní již jen potvrdit, a pokud vše funguje správně, mělo by se aktivovat tlačítko „AT Mode“. To přepíná modul do režimu pro nastavení.

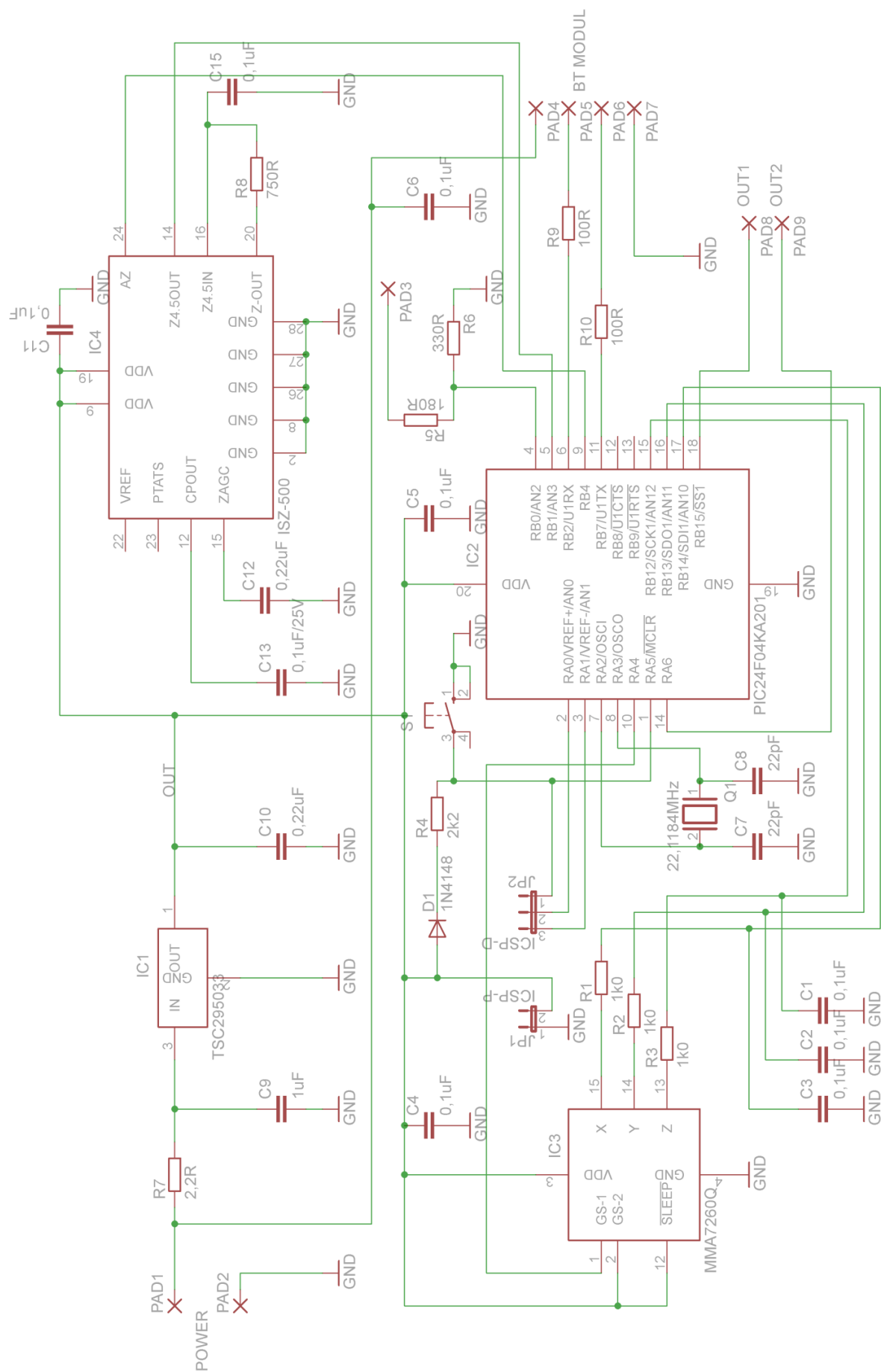
V nastavujícím režimu je dobré nejdříve povolit konfiguraci modulů pomocí Bluetooth. Dále je nutné nastavit moduly tak, aby je bylo možné vyhledat a spárovat. Heslo pro párování se zadávat nemusí, já jsem ho však zadal, a to pro většinu zařízení základní „0000“. Po každém změněném nastavení je dobré toto zapsat i do modulu tlačítkem „Write“. Nyní je možné se odpojit tlačítkem „Disconnect“. Konfigurace modulů přímo pomocí Bluetooth má nespornou výhodu v tom, že při nastavení komunikační rychlosti vyšší než podporuje sériový port počítače a i jeho Bluetooth modul, se moduly konfigurací „přes vzduch“ dokáží připojit na své základní rychlosti (57600 baudů za sekundu) bez ohledu na rychlost skutečně nastavenou. Díky této funkci jsem byl schopen nastavit relativně vysoké přenosové rychlosti i přes to, že sériový port a vestavěný Bluetooth mého počítače podporují rychlost jen 128000 baudů za sekundu. Ostatní parametry jsou však již platné a musí se vložit správně. Také se vždy při této metodě konfigurace povoluje řízení toku pomocí CTS/RTS.

Pro připojení komunikačního modulu pomocí v počítači vestavěného Bluetooth rozhraní je nutné odpojit pouze sériový konektor z počítače, ale USB napájení ponechat. Poté se ovládacím software Bluetooth v počítači nalezne nové zařízení – komunikační modul. Po zadání hesla a úspěšném spárování je nutné otevřít nastavení takto spárovaného transcieveru a zjistit, který virtuální sériový port mu byl přidělen. Tento se poté vybere v příslušném okénku nastavení komunikace v programu „Serial Port Adapter Toolbox“. Po opětovném vstupu do konfiguračního módu jsem na kartě „Serial“ rychlost změnil na 921,6 kbps, paritu nastavil na lichou (odd) a počet stop bitů na 2. Samotné spárování 2 modulů je popsáno na



stránkách distributora [19], ale já ho zde pro úplnost zprávy také uvedu. Přešel jsem tedy na kartu „Client“, kde jsem zvolil SPP profil pro komunikaci a počet připojených zařízení (number of remote peers) na 1. Poté jsem nastavení zapsal to modulu tlačítkem „Write“. Po připojení druhého modulu k napájecímu napětí, jsem stisknutím „Device discovery“ začal jeho hledání. Po nalezení druhého modulu jsem nastavil jeho identifikační číslo (peer id) na 0. Zatrhnutím „Always connected“ jsem určil, aby moduly byly vždy spojeny a z okénka „Search for and select remote peer“ vybral jeho adresu. Tímto je spárování dokončeno, a pokud budou mít oba transreceivery připojené napájení, budou spojeny. Navíc tím, že jsem nastavil jen jedno připojitelné zařízení, nebude možné se k nim ve spojeném stavu připojit a tak se do jisté míry chrání proti přepsání nastavení. A zároveň není nutné v nastavení vypínat funkce nalezení a spárování, díky čemuž by se moduly v případě změny nastavení musely připojovat metalicky pomocí sériového portu. Takto se pouze odpojí napájení od jednoho z modulů a tím se k druhému (samozřejmě se znalostí hesla) lze připojit a změnit jeho nastavení.

Na obr. 4.8 je schéma modulu pro letadlo. Už z počátku byl vybrán řídicí mikrokontroler (IC2) PIC24F04KA201. Je to hlavně díky mým zkušenostem s těmito jednočipy. Tento konkrétní je pro úlohu zcela ideální, není nijak složitý a má téměř jen ty potřebné funkce. Akcelerometr (IC3) MMA7260 byl též už od počátku zahrnut do návrhu odměřování zrychlení. Je to hlavně díky jeho nenáročnosti a dostupnosti u nás. Akcelerometr jsem připojil k integrovanému analogovo-digitálnímu převodníku v mikrokontroleru. Měření rozsah je nastaven na ± 4 g, aby se nechala změřit odstředivá síla. Jednočip sbírá data z jeho tří os. Než jsou však data zpracována mikrokontrolerem procházejí filtry z rezistorů R1 – R3 a kondenzátorů C1 – C3. Tyto filtry pomáhají potlačit rušení a šum, který by mohl vycházet z akcelerometru. Celé zapojení je napájeno ze zdroje, který je společný i pro výkonovou část. Toto napětí o velikosti 5 V se přivádí pomocí kroužků, v otočném uložení úlohy, nejprve k Hallově snímači a poté dál až k měřicímu modulu. Bluetooth transceiver má svůj integrovaný lineární stabilizátor a tedy je napájen přímo tímto napětím. Ostatní součástky však vyžadují 3,3V napájení. To je zabezpečeno lineárním stabilizátorem IC1. Jedná se o typ TSC295033, který je v pouzdře TO92. Maximální zatížení je 100 mA, takže je dostatečně dimenzován.

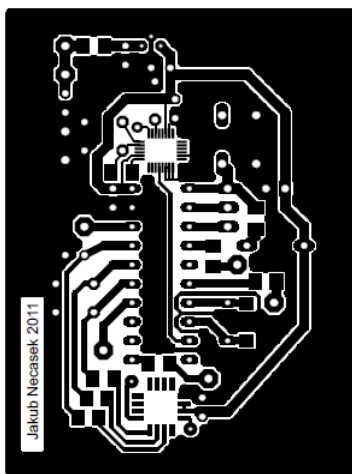


Obr. 4.8: Schéma zapojení modulu pro letadlo

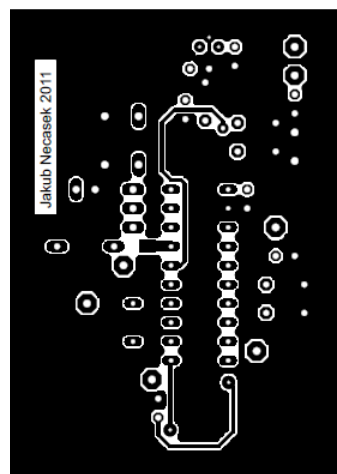


Před tímto stabilizátorem je vložen filtr typu dolní propust sestávající z rezistoru R7 a kondenzátoru C9. Tento filtr má za úkol odstranit vysokofrekvenční rušení, na které je zvláště gyroskop náchylný. Mikrokontroler má vyvedeny piny pro ICSP programování, jedná se o JP1, na který se připojuje napájení programátoru a JP2, který obsahuje programovaná data. Vzhledem k programování jednočipu vyšším napětím na pinu MCLR (master clear) než jaké je v obvodu, je zde dioda D1, která zabraňuje, aby se toto napětí při programování dostalo na ostatní součástky. Dále je na tento vstup připojeno tlačítko S1, jehož stisknutím se MCLR zkratuje na zem a tím se běh programu v mikrokontroleru resetuje. Výstup Hallova snímače polohy se připojuje na pájecí plošku PAD3. Z ní poté analogový signál prochází děličem, který ho přizpůsobuje napětí obvodu. Tady se signál 0 – 5 V sníží na úroveň 0 – 3,3 V pro vyhodnocení analogovo-digitálním převodníkem. Pájecí plošky PAD4 – PAD8 slouží k připojení Bluetooth modulu. Elektronický gyroskop IC4 byl doplněn až těsně před výrobou plošného spoje, po dohodě s vedoucím práce. Je připojen také k analogovo-digitálnímu převodníku mikrokontroleru. Jeho výstup Z-OUT který má rozsah ± 500 °/s, je veden přes filtr zpět do pouzdra, ve kterém je integrován zesilovač se zesílením 4,5 pro dosažení výsledného rozsahu ± 110 °/s. Na výstupech OUT1 a OUT2 (pájecí plošky PAD8 a PAD9) je zpětný signál od počítače k letadlu.

Na obr. 4.9 je spodní strana a na obr. 4.10 vrchní strana desky plošných spojů pro tento modul. Obrázky jsou ve skutečné velikosti. Jejich návrh je podřízen nároku na minimální velikost, výměnný mikrokontroler a odolnost proti rušení. Z důvodu minimalizace rušení, je tedy použita technika rozlité mědi na co největší ploše tištěného spoje. Plocha mědi je poté připojena k zápornému pólu napájecího napětí.



Obr. 4.9: Spodní strana desky pro letadlo



Obr. 4.10: Vrchní strana desky pro letadlo



Aby byla zaručena vždy co nejkratší cesta signálu, jsou na několika místech výběžky rozlité mědi propojeny se zbytkem stínění tenkými propojkami například mezi vývody mikrokontroleru. Speciálním stíněním byl opatřen obvod krystalového oscilátoru, aby tak nezarušoval ostatní signály na desce. Velkou nevýhodou akcelerometru a elektronického gyroskopu je nemožnost pod jejich pouzdry vést cesty. Toto je zapříčiněno tím, že zespodu pouzdra je odhalená kovová destička, na které je připevněn samotný křemíkový čip. Vedením cest pod součástkou by se tyto mohly s destičkou zkratovat a zničit součástku samotnou nebo okolní. Díky oboustrannému plošnému spoji je maximalizována minimalizace celého návrhu. Celkem byly potřeba jen tři průchodky signálu z jedné strany na druhou. Tímto ale vyvstává nutnost pájet vývody většiny THT součástek z obou stran desky. V případě použití čistě SMT součástek, by se mohla celá deska plošných spojů mnohonásobně zmenšit.

Desku plošných spojů jsem vyráběl v „elektro-bastlárně“ pana Ing. Lubomíra Slavíka v budově E Technické Univerzity. Motiv pro osvětlování jsem vytiskl na průhlednou fólii. Tu jsem zvolil z důvodu nejlepšího kontrastu, který je u takto malých cest a pájecích plošek potřeba.

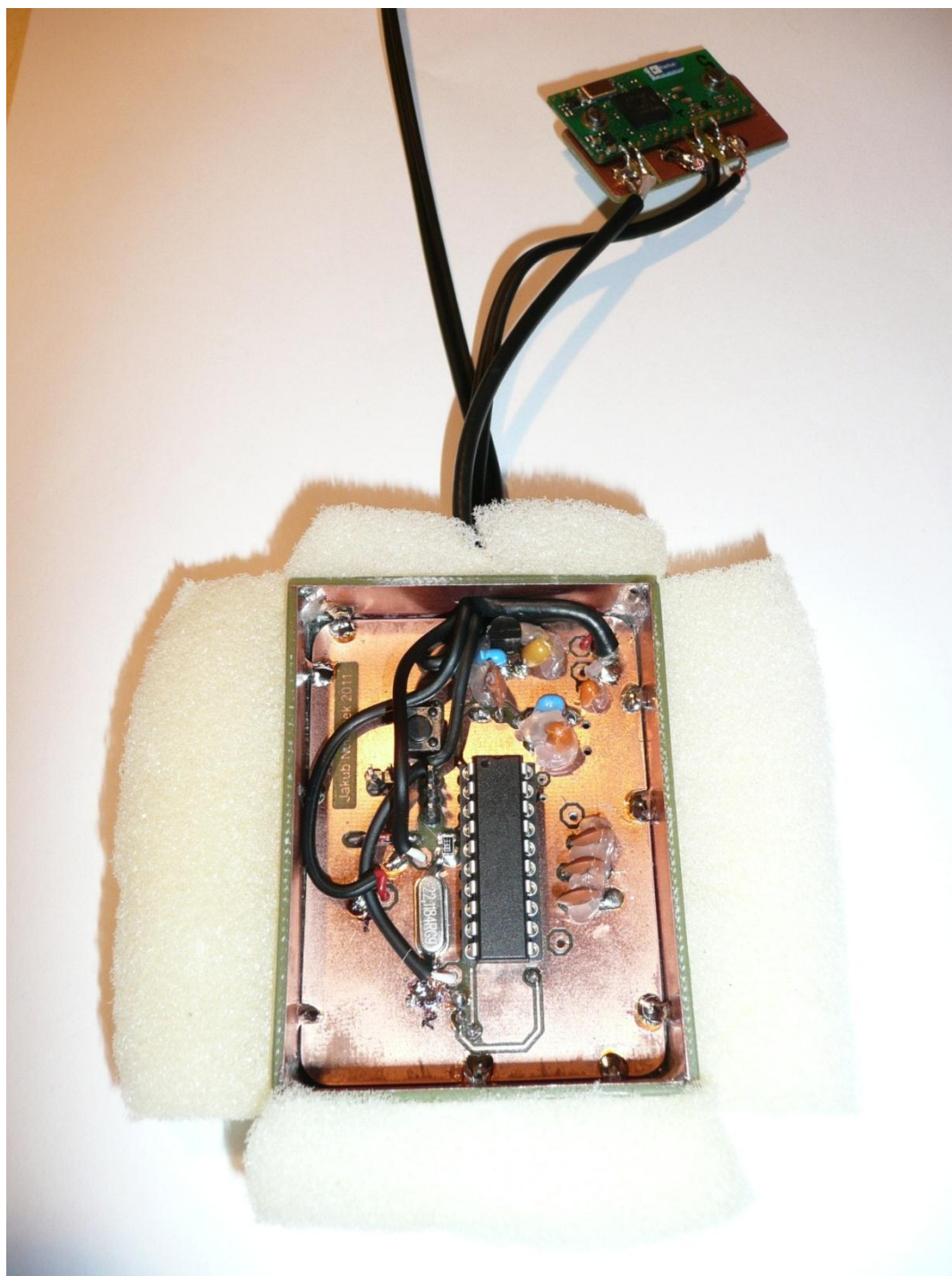
Osazovat jsem začal s ohledem na velikost nejdříve elektronický gyroskop. Protože se plošky takto blízko u sebe nevyleptaly, bylo nutné je od sebe odfrézovat. Plošky jsem poté pocínoval, přiložil gyroskop a zahřál vývody tak, aby se slilo pocínování na pouzdru s tím na plošce. V pořadí druhou pájenou součástkou byl akcelerometr vzhledem k jeho jen o málo větším rozměrům. Tento se již pájel dobře, protože disponuje mnohem méně vývody, které jsou tak větší. Následně jsem osadil všechny SMT součástky a takto pokračoval dále. Za zmínku stojí upevnění krystalového oscilátoru, který má nejen připájené vývody, ale také pouzdro – tím je dokonale stíněn a zároveň velmi pevně drží na svém místě. Nakonec jsem připevnil všechny kondenzátory k desce tavnou pistolí, aby se vibracemi neulomily. V příloze A (s. 56) je obrázek desky s cestami i osazenými součástkami, v příloze B (s. 57) je seznam součástek.

4.6 Instalace modulu do makety letadla

Po osazení a oživení modulu, jsem přistoupil k výrobě krabičky, do které bude uložen. Vzhledem k nestandardním rozměrům, jsem se rozhodl pro krabičku z cuprexitu vyrobenou přímo pro tento modul. Takto vyrobená krabička má i nespornou výhodu dobrého stínění svého obsahu. Dále je velmi pevná a snadno vyrobitelná. Po nastříhání a spájení pruhů



cuprextitu, stačilo jen připasovat desku modulu a nakonec jí dovnitř zapájet. Bluetooth modul má též pevný základ tvořený destičkou cuprextitu, ke které je přišroubován a přes drátky připájen. Toto uložení dovozuje mnohem lepší připevnění datových a napájecích kabelů. Pro napájení modulu byl použit koaxiální kabel a pro data stíněná dvoulinka. Takto jsem se rozhodl z důvodu minimalizace rušení komunikace transrecieveru. Celou krabičku jsem před instalací do letadla ještě oblepil molitanem z důvodu utlumení vibrací tvořených motorem. Na obr. 4.11 je již hotové uložení i s připojeným Bluetooth modulem a molitanovou izolací.



Obr. 4.11: Hotové zapouzdrění měřicího modulu před instalací do letadla



Po dokončení zapouzdření měřicího modulu, jsem přistoupil k úpravám na letadle samotném. Nejprve bylo třeba zvolit správné místo pro uložení krabičky s elektronikou. Ta vzhledem k osazení snímači musí být na výšku a zadní stranou (s SMT součástkami) dopředu. Bohužel takováto orientace je jedna z nejhorších možných vzhledem k instalaci do makety letadla. Pro jinou orientaci, by ale bylo nutné každý snímač umístit na jeho vlastní destičku a připájet je kolmo k desce plošných spojů. Takovéto alternativní umístění sebou přináší mnoho problémů. Prvním je správné geometrické umístění. Dalším je zpevnění takovéto konstrukce proti kmitům. A v neposlední řadě by takto kolmo umístěné destičky se snímači zvětšily rozměry krabičky natolik, že by se to v konečné podobě ani nemuselo vyplatit. První operací na maketě letadla bylo vyříznutí otvoru pro vložení krabičky. Tento otvor je vidět na obr. 4.12 a jedná se o ten levý.



Obr. 4.12: Vyříznutý otvor v letadle pro zasunutí krabičky

Dále jsem vyhloubil otvor v protilehlé stěně, který bude krabičku držet na místě. Bohužel není příliš hluboký vzhledem k tenkému polystyrenu tvořícímu stěnu. Pro uvolnění místa kabelům vedoucím ke kroužkům, jsem musel ustříhnout silové vodiče motoru, který je v letadle. Tento je však již dávno odpojen a nepoužívá se pro jeho nízký výkon. Z letadla bohužel vymontovat nelze – muselo by se celé po délce rozříznout, protože je do něj



montován před slepením. Po protažení vodičů trubičkou, na které celá maketa visí, jsem smáčknul molitan blíže ke krabici a vložil jí do připraveného výřezu. Po umístění se molitan rozepřel v trupu letadla. Tímto je provedena fixace krabice v trupu a zároveň se tak brání v přenosu vibrací. Vpravo na obr. 4.12 je také vidět prolis ve tvaru „V“ v polystyrénových bočnicích. Do tohoto prolisu jsem zaklínil Bluetooth transceiver s jeho vyztužující destičkou (v tomto pohledu) vzhůru. Je tak vzdálen od motoru a hlavně od odstíněné krabice s elektronikou, která by mohla zmenšit jeho dosah. Finální operací na maketě bylo opětovné vlepení vyříznutého obdélníku. Tímto je krabice s elektronikou upevněna v letadle.



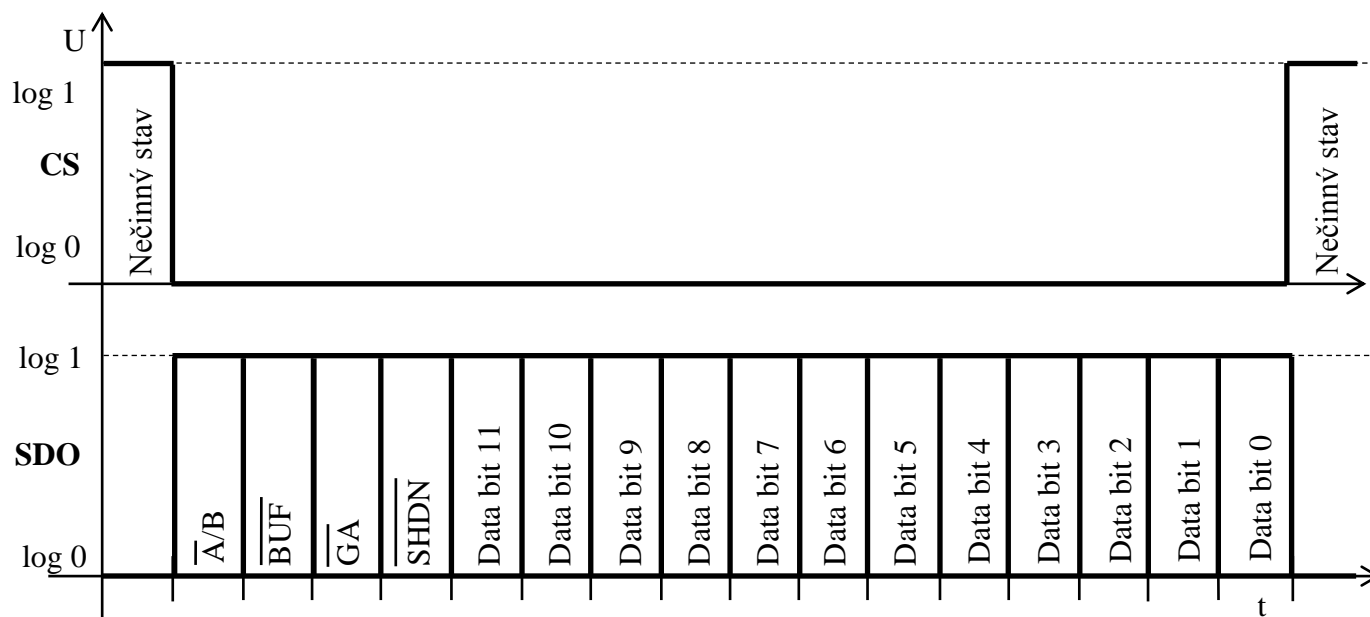
5 Popis vybraného řešení – modul pro převod dat

5.1 Digitálně-analogový převodník

Pro převod digitálního signálu z mikrokontroleru, slouží digitálně-analogový převodník (D/A převodník). Vyráběny jsou společností Microchip stejně jako osazené mikrokontrolery. Jedná se o dvojité 12bitový převodník MCP4922 [20] v jednom pouzdře DIP14. Dovoluje napájení od 2,7 do 5,5 V. Obsahuje vstup pro externí napětíovou referenci pro každý převodník zvlášť. Převodník disponuje mnoha užitečnými funkcemi, jako je možnost synchronizovat výstupní napětí všech převodníků tak, aby se měnilo v přesně daný okamžik, nebo možnost vypnout po sběrnici jednotlivé výstupy, případně oba převodníky pomocí jednoho z vývodů. Vzhledem k náročnosti na vstupy pro dvakrát 12 bitů, je převodník opatřen sériovou sběrnicí SPI (Serial Peripheral Interface). Tuto obsahuje i zvolený mikrokontroler.

SPI je typu Master-Slave. Master generuje hodinový takt všem připojeným zařízením a také vybírá, se kterým bude komunikovat. Hodinový takt je však generován jen při komunikaci a to ve stejném počtu jako posílaných bitů. Sběrnice je založena na kruhovém zásobníku. Pro odesílání dat ze strany Master stačí jednoduše zapsat data do výstupního zásobníku, který je odešle. Zároveň s každým odeslaným bitem však také naslouchá, co vysílá připojené Slave zařízení a ukládá data zpátky na místo odeslaných. Z tohoto je tedy zřejmé, že pro přijetí dat z připojených zařízení je nutné do výstupního zásobníku zapsat náhodné číslo, které se odešle, a tím se přijmou žádaná data. Pro komunikaci využívá vývodů SDO (serial data out – výstup dat), SDI (serial data in – vstup dat) a SCK (serial clock – hodinové pulsy).

Každé SPI zařízení má jinou podobu posílaného paketu – to je dáno hlavně různými připojitelnými zařízeními na tuto sběrnici, která mohou mít různé specifické funkce. Ale vždy je tento paket podrobně popsán v datovém listu. Já zde jeho popis pro úplnost též uvedu. Vybraný digitálně-analogový převodník komunikuje pomocí 16bitového datového paketu. Pro odeslání dat je nejprve nutné zvolit připojené zařízení unikátním CS (chip select) signálem. Tento je invertován, tedy v klidu má hodnotu napětí blízkou napájecímu. Po zvolení zařízení nastavením log. „0“ na jeho CS vstupu může komunikace začít posláním konfiguračních bitů následovaných datovými. Na obr. 5.1 je znázorněn datový paket komunikace zvoleného převodníku i s řídícím CS signálem.



Obr. 5.1: SPI komunikace s převodníkem

První bit je výběr, pro který převodník jsou data určena. Druhý nastavuje, zdali má být napěťová reference na vstupu V_{ref} , podpořena vnitřním zásobníkem. Předposlední konfigurační bit nastavuje volitelné zesílení výstupu, kdy převodník bude používat buď celý svůj rozsah 4096 hodnot, nebo jen poloviční, ale při stejném výstupním rozsahu. Tím se výstup dvojnásobně zesílí. Poslední bit konfigurace určuje, zda má být převodník aktivní nebo v úsporném režimu s nulovým výstupem (vypnut). Toto nastavení následuje 12 datových bitů, jejichž hodnota se ihned po opětovném navrácení CS signálu do log. „1“, zapíše na výstup. V případě potřeby synchronizovat výstupy převodníků by se tento signál (též invertovaný jako CS) připojil na vstup LDAC.

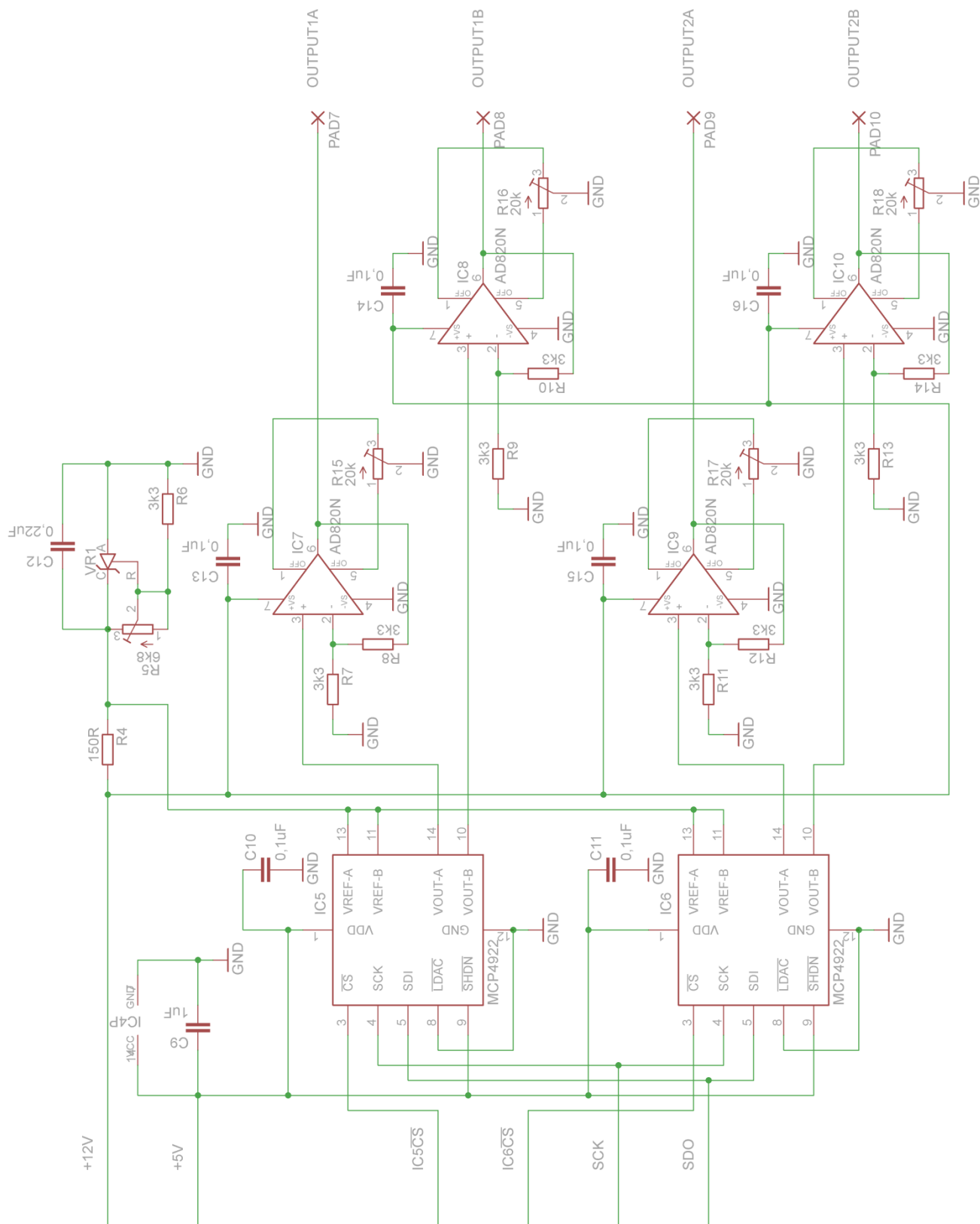
5.2 Operační zesilovač

Vzhledem k napájecímu napětí digitálně-analogového převodníku maximálně 5 V bylo nutné jeho signál zesílit na hodnotu 0 – 10 V. K tomu je zapojení vybaveno operačním zesilovačem od společnosti Analog Devices. Jedná se o AD820 [21], rail-to-rail (jeho výstup je v celém rozsahu napájecího napětí) operační zesilovač v pouzdře DIP8. Je určen pro nesymetrické napájení a obsahuje vývody pro kompenzaci napěťové nesymetrie.

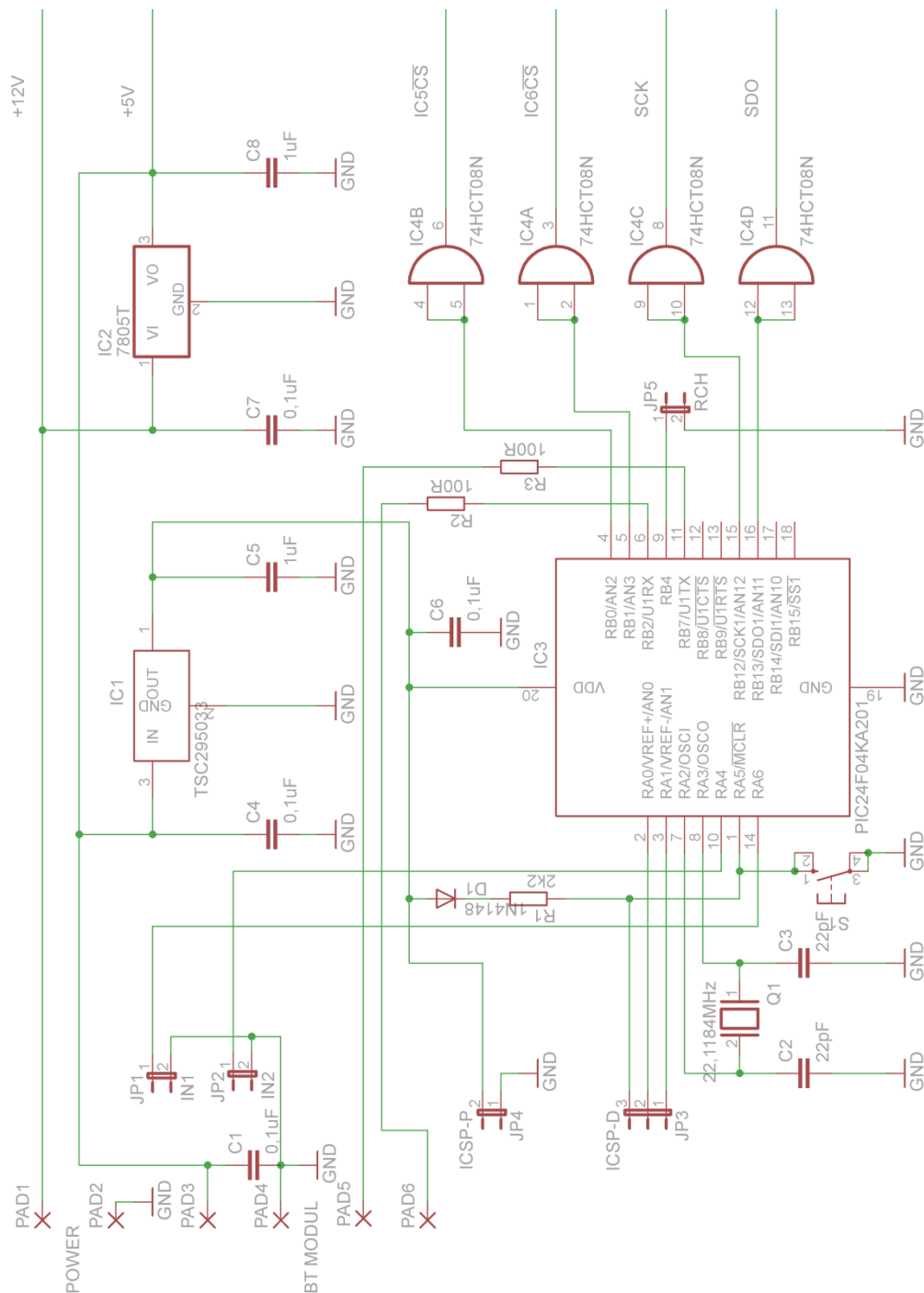


5.3 Popis vývoje a funkce zapojení převodníku pro měřicí kartu

Nyní bych rád popsal vývoj a funkci zapojení převodníku signálu pro měřicí kartu. Srdcem obvodu je mikrokontroler PIC, stejný jako u předchozího modulu. Také Bluetooth transceiver je shodný. Stačilo tedy vybrat vhodný digitálně-analogový převodník, který by naměřený a digitalizovaný signál opět sestavil do jeho původní analogové podoby. Nakonec jsem vybral operační zesilovač, který tento signál přizpůsobí kartou měřenému rozsahu. Na dalších dvou stranách je rozděleno schéma tohoto modulu. Na obr. 5.2 je první a na obr. 5.3 druhá část schématu. Na druhé části je vidět vstup pro stabilizované napětí 12 V ze spínaného zdroje, které napájí přesnou napěťovou referenci s obvodem TL431 (VR1), operační zesilovače a 5V lineární stabilizátor 7805T (IC2). Napětím ze stabilizátoru IC2 se napájí Bluetooth modul, digitálně-analogové převodníky (IC5 a IC6), integrovaný obvod 74HCT08D (IC4) a také se dále upravuje stabilizátorem TSC295033 (IC1). Z tohoto stabilizátoru vychází napětí 3,3 V pro napájení mikrokontroleru PIC (IC3). Digitálně analogové převodníky by mohly pracovat na napětí 3,3 V, ale vzhledem k počtu kroků (4096), by se na tomto napěťovém rozsahu mohla tvořit chyba. Proto jsem pro převodníky zvolil plné 5V napájení. Abych zajistil spolehlivou komunikaci mikrokontroleru s převodníky, použil jsem stejný způsob uzpůsobení úrovní jako u převodníku RS232 na UART. Tedy pomocí AND hradel se spojenými vstupy. Proto je zde integrovaný obvod 74HCT08D. Zapojení mikrokontroleru je shodné s tím v modulu pro letadlo. Má vyvedeny vstupy pro ICSP programování (JP3 a JP4), osazen stejný krystalový oscilátor s hodnotou 22,1184 MHz. Vstup resetování programu MCLR je též shodně zapojen. IN1 a IN2 (JP1 a JP2) jsou vyvedené vstupy jednočipu pro signál jdoucí zpátky na letadlo – odpovídají výstupům OUT1 a OUT2. Vstupem RCH se v měřícím modulu přepínají posílaná data, ale o tom až v popisu programu. Po opětovném sestavení analogového průběhu digitálně-analogovými převodníky, se tento zesiluje operačními zesilovači (IC7 – IC10). Ty jsou zapojené jako neinvertující zesilovače se zesílením 2. Pro zpětnou vazbu byly ohmmetrem vybrány rezistory se stejnou hodnotou odporu za účelem co nejpřesnějšího nastavení zesílení. Odporové trimry R15 – R18 jsou zde za účelem eliminace vstupní napěťové nesymetrie operačních zesilovačů.



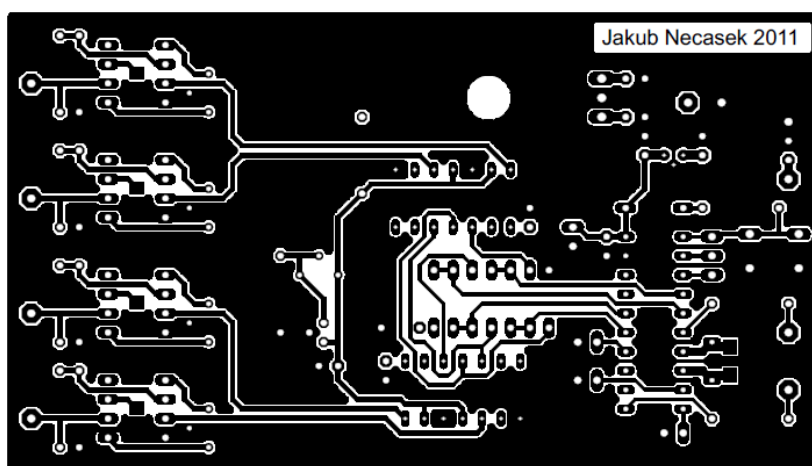
Obr. 5.2: První část schématu převodníku pro měřicí kartu



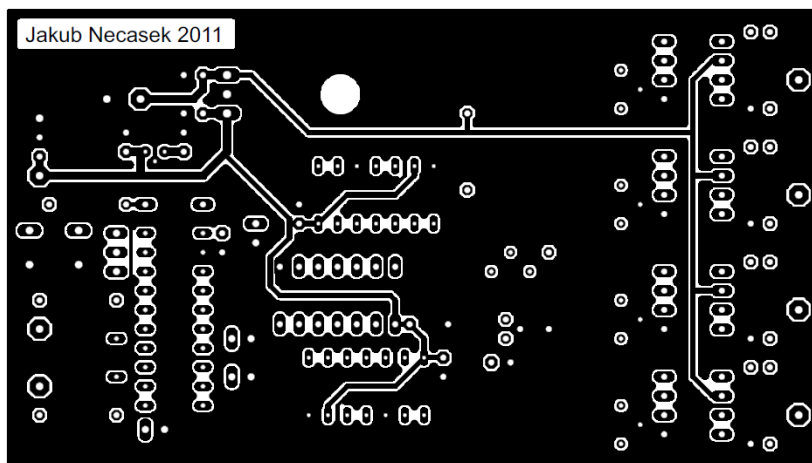
Obr. 5.3: Druhá část schématu převodníku pro měřicí kartu



Při návrhu desky plošných spojů je opět kladen důraz na použití vyměnitelných součástek v patičkách a na stínění před rušením. Plošný spoj je navržen jako oboustranný. Spodní strana nese všechny signálové vodiče, vrchní poté většinu napájecích. Je zde také použita metoda rozlité mědi pro odstranění smyček, do kterých by se mohlo indukovat rušení. Stejně jako u předchozí desky jsou na mnoha místech dodatečně vloženy propojky rozlité mědi za účelem co největšího zkrácení cesty signálu. Také vývody krystalového oscilátoru mají své speciální stínění. A stejně jako u předchozího modulu je pouzdro krystalu připájeno k ploše rozlité mědi. Na obr. 5.4 je spodní strana desky plošných spojů a na obr. 5.5 je strana vrchní. Obrázky jsou ve skutečné velikosti.



Obr. 5.4: Spodní strana desky pro převodník



Obr. 5.5: Vrchní strana desky pro převodník

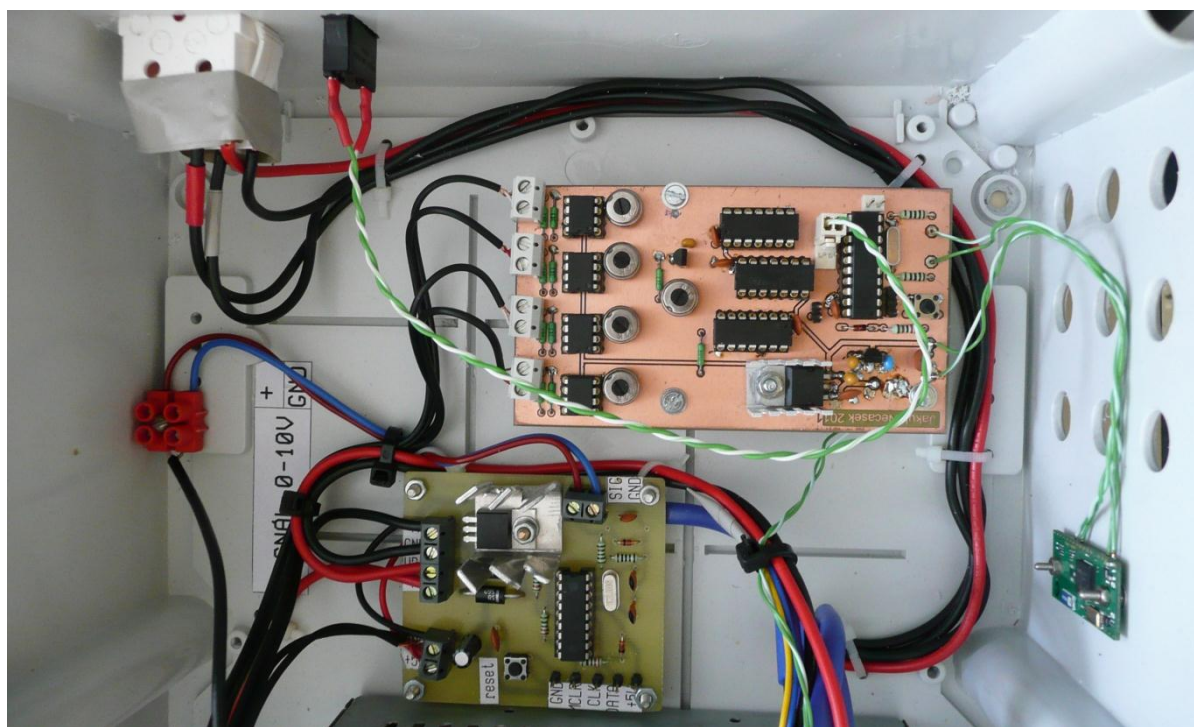
Desku plošných spojů jsem vyráběl zároveň s deskou pro modul v letadle. V příloze C (s. 58) je velký obrázek desky s cestami i součástkami a v příloze D (s. 59) seznam součástek. Fotografie hotové desky je součástí následující kapitoly o instalaci modulů do učebny.



6 Instalace modulů do učebny

6.1 Instalace převodníku pro měřicí kartu

Pro vestavění převodníku do centrální krabice úlohy bylo nutné nejdříve tuto odšroubovat od stolu. Deska je uchycena pomocí distančních sloupků M3 do dna krabice. Vodiče napájení a koaxiální kabely pro vedení signálu k měřicí kartě jsou upevněny ve šroubovacích svorkách na plošném spoji. Tato konstrukce je vybrána za účelem snadné demontáže. Bluetooth modul je přišroubován ke spodní stěně krabice pomocí šroubků M2. Jeho napájení a datové vodiče jsou k desce převodníku napevno připájeny. Přepínač snímačů je k desce připojen konektorem s pojistným háčkem. Poslední operací bylo doplnění vodičů ke kartě o ochranné rezistory a transily na 12 V. Ty by měly omezit proudové a napěťové špičky, které by mohly vzniknout na vedení a poškodit tak měřicí kartu v počítači. Na obr. 6.1 je vidět celé zapojení převodníku již zabudované v centrální skříni. Deska na obrázku dole je modul řídicí výkon motoru.

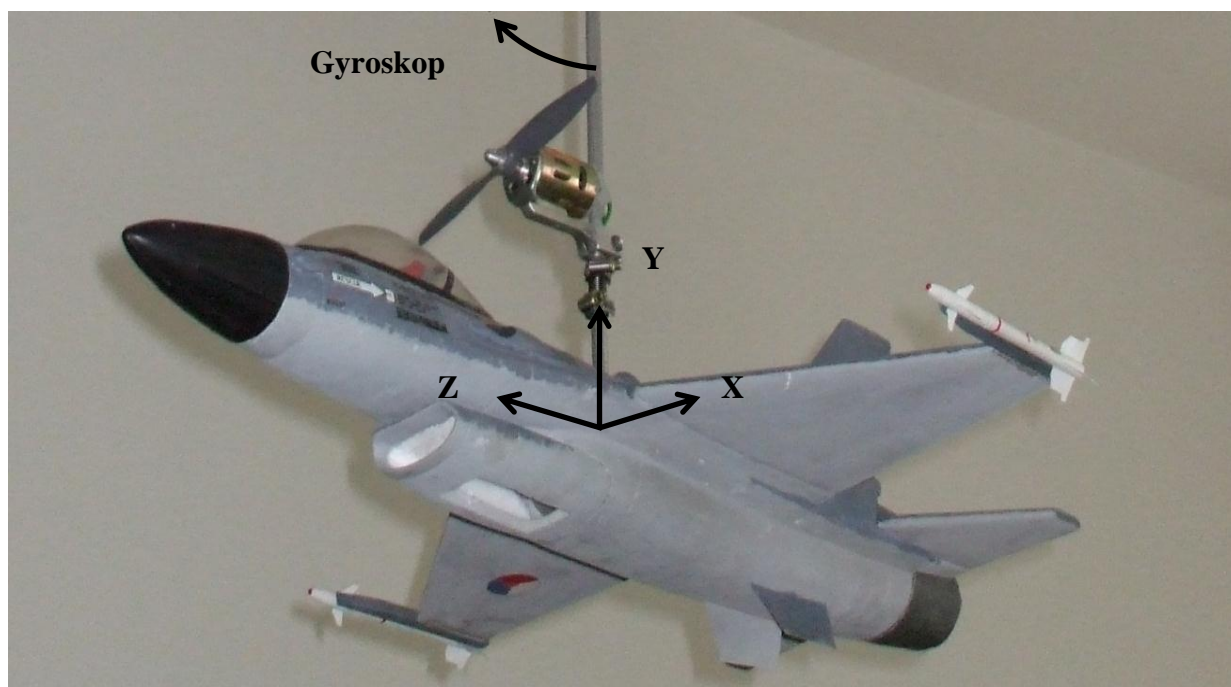


Obr. 6.1: Zabudované zapojení převodníku pro měřicí kartu



6.2 Instalace letadla na závěs

Po zabudování krabičky s elektronikou do letadla a převodníku do krabice jsem vyměnil stávající silové vodiče, přivádějící napětí svislou trubkou k motoru, za nové. Spolu se silovými vodiči byla k Hallova snímači, umístěném v lomení ramen, přivedena také stíněná dvoulinka. Ta obsahuje napájení měřicího modulu a také analogový signál z Hallova snímače. Po protažení vodičů jsem připevnil maketu letadla zpátky na její nosnou tyč. Upevnění je řešeno pomocí pružiny. Tato má funkci ochrany závěsu letadla, například při nechtěném zavadění procházejícími osobami. Na obr. 6.2 je souřadný systém letadla s naznačením kladných směrů os – kladného napětí ze snímačů při pohybu tímto směrem. Na výstupu Hallova snímače se při odklonu zmenšuje napětí.

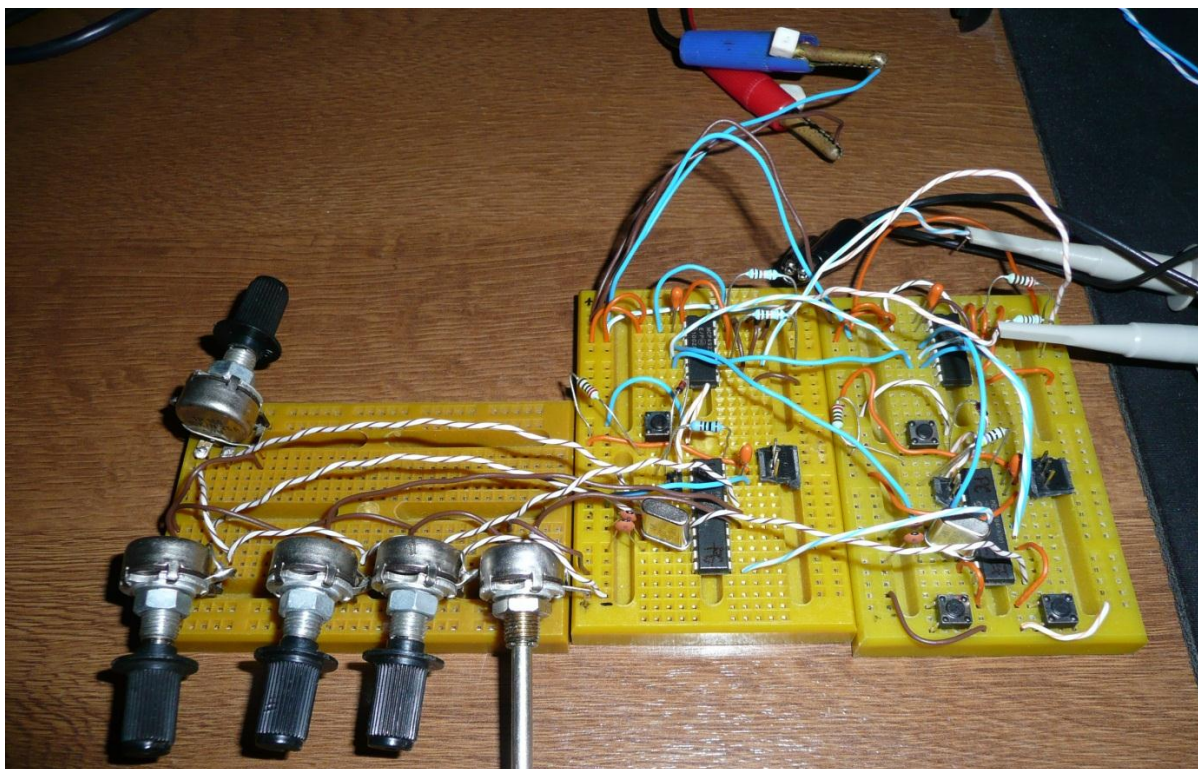


Obr. 6.2: Souřadný systém letadla



7 Vývoj a popis programu mikrokontroleru

V této kapitole se budu věnovat vývoji a funkci programu v jednotlivých mikrokontrolerech. I přes mé zkušenosti s těmito jednočipy byl vývoj programu pro ně jednou z nejdelších a nejsložitějších prací na této úloze. Programy jsem psal v jazyku C do editoru přímo od výrobce jednočipů – Microchip. Editor se jmenuje MPLAB IDE a mnou používaná verze byla 8.63. Abych mohl programovat vybraný jednočip v jazyku C, bylo nutné do editoru vložit kompilátor MPLAB C30 pro programování 16bitových rodin PIC24, dsPIC30F a dsPIC33F. Editor MPLAB IDE je ke stažení zdarma přímo ze stránek výrobce. Avšak kompilátor C30 je placený produkt. Naštěstí výrobce umožňuje po zaregistrování stáhnout studentskou verzi LITE, ve které sem pracoval. Pro ověřování funkčnosti programu jsem postavil na nepájivém poli zjednodušené zapojení obou modulů. Bluetooth moduly jsem nahradil vodiči – což je i jejich úkol. Toto zapojení je vidět na obr. 7.1.



Obr. 7.1: Testovací zapojení na nepájivém poli



7.1 Program pro modul v letadle

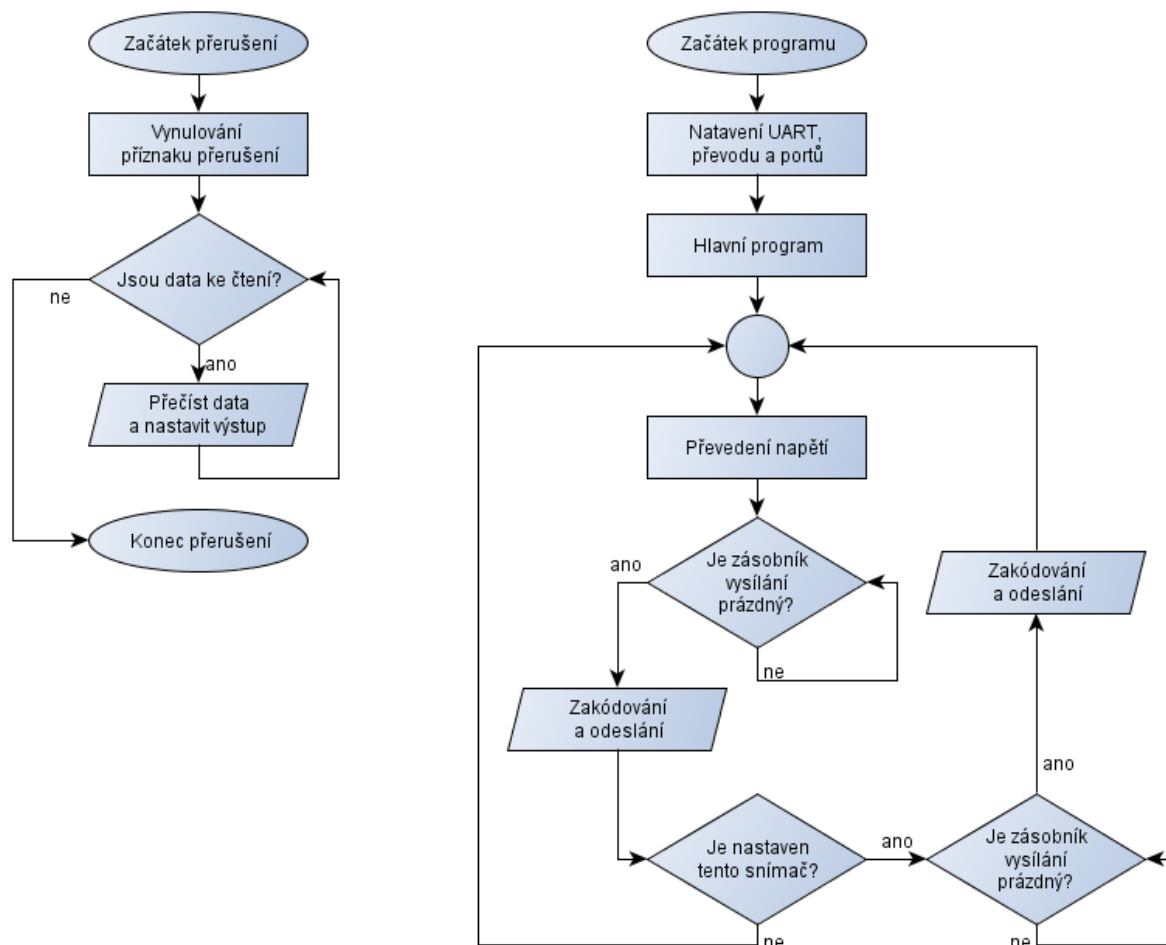
Tento program má za úkol změřit všech pět analogových napětí na vstupech a odeslat je pomocí UART do Bluetooth modulu. A zároveň číst příchozí data a podle nich nastavovat výstupy a posílané hodnoty. Proč posílané hodnoty – snímače jsou tři, ale akcelerometr má tři výstupy. K vyhodnocení výsledných pěti signálů by byl zapotřebí ještě jeden převodník pro jejich odeslání do měřicí karty. Při konzultaci s vedoucím práce jsme se však shodli, že 4 měřené signály najednou naprosto dostačují. Proto se osa Z akcelerometru přepíná s výstupem elektronického gyroskopu. Toto se provádí změnou polohy přepínače na centrální krabici úlohy. Takto se i ušetří čas pro odesílání pátého náměru. Výpis celého zdrojového kódu programu pro letadlo je pro svůj rozsah v příloze E (s. 60). Program je popsán v angličtině, aby i nečesky hovořící čtenáři mohli z této práce čerpat.

Základním problémem při tvorbě programu byla nekompatibilita měřených a posílaných dat. Zatímco analogovo-digitální převodník udává výsledek na 10 bitech, UART linka dokáže najednou odeslat jen 8 bitů dat. Navíc je ještě potřeba přenést informace o původu dat. Tento problém jsem vyřešil rozdělením dat na polovinu (5 bitů), a tím vzniklo místo o velikosti tří bitů pro informaci o původu dat. Prvního z těchto tří bitů jsem využil pro informaci o první nebo druhé polovině dat. Ve druhých dvou bitech je zakódován jejich původ a tím i cíl. V tab. 7.1 je postup zakódování názorně předveden.

Tab. 7.1: Kódování dat modulem v letadle

Z prvního vstupu AD převodníku: 1010101010	
První polovina dat: 1 00 10101	Druhá polovina dat: 0 00 01010
Z druhého vstupu AD převodníku: 1010101010	
První polovina dat: 1 01 10101	Druhá polovina dat: 0 01 01010

Běh samotného programu začíná nastavením UART modulu. Poté je nastaven analogovo-digitální převodník a poslední konfigurací jsou směry portů (vstup/výstup). Podrobně tu rozebírat každé nastavení zvlášť nemá smysl. Vše je zřejmé z popisů v příloženém zdrojovém kódu. Na obr. 7.2 je vývojový diagram znázorňující běh hlavního programu a obsluhy přerušení od příjmu dat.



Obr. 7.2: Vývojový diagram programu pro letadlo

Tento vývojový diagram neobsahuje popis celého programu, ale jen několik dílčích bloků pro názornost. Po nastavení všech parametrů periferií, začne hlavní program. Ten nejprve převede napětí ze všech pěti vstupů. Následně je testováno, zdali je volný zásobník modulu UART pro odeslání dat. Pokud tomu tak je, tak se první polovina dat výše uvedeným způsobem zakóduje a odešle. V případě, že ne, tak v tomto testování čeká do uvolnění místa. Poté se opět testuje volné místo a tak dále. Snímače, které je možno přepínat před testováním zásobníku testují ještě nastavení svého zvolení.

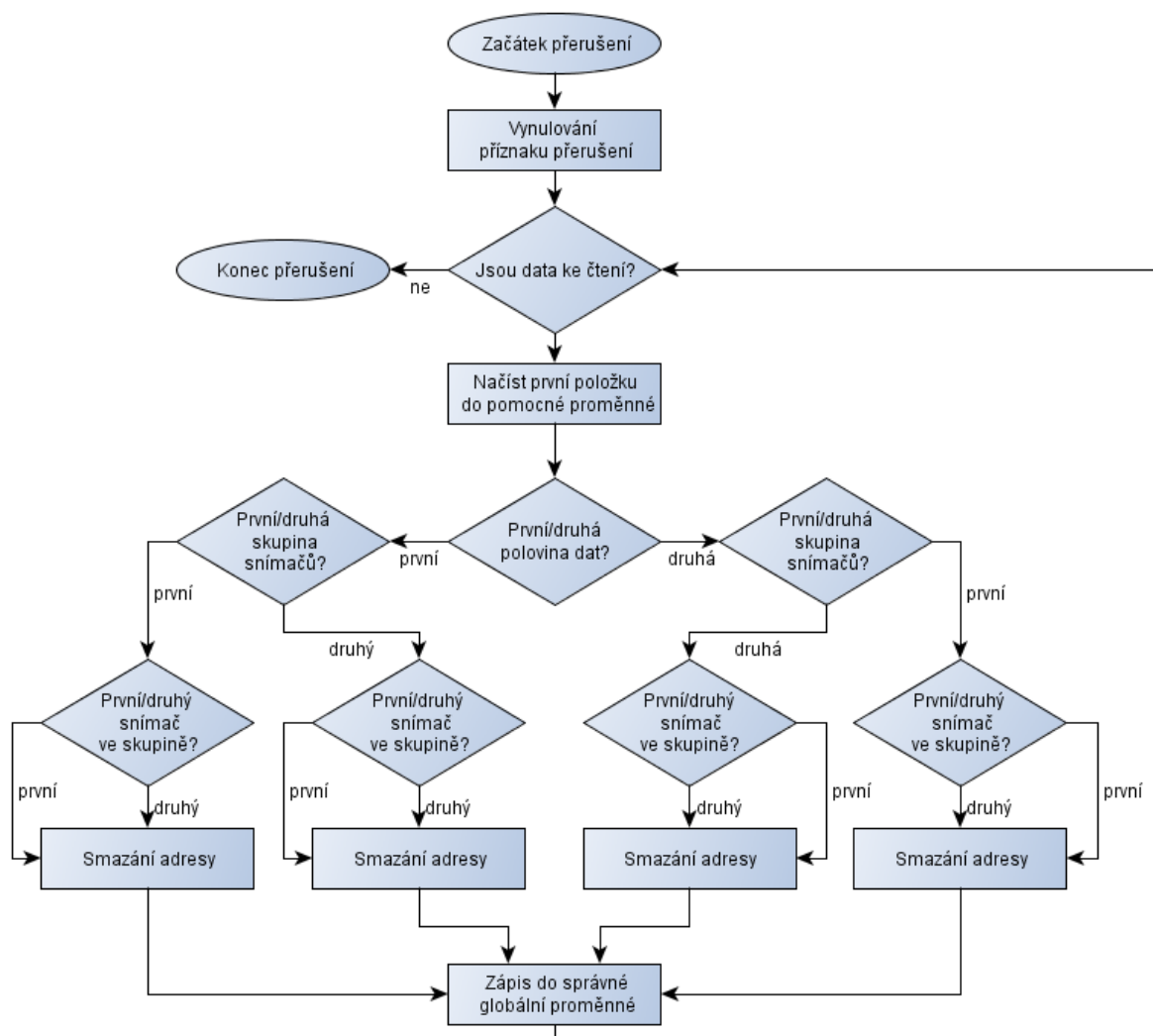
V obsluze přerušení od příjmu dat modulem UART, se nejdříve vynuluje příznak přerušení, aby byl možný návrat do hlavního programu. Poté se testuje, zdali jsou připravena data ve vstupním zásobníku. Pokud ano, data se přečtou do pomocné proměnné (přečtením se zásobník posouvá) a vyhodnotí se. Takto se nastavují dva zpětné signály z počítače do letadla OUT1 a OUT2. Též se tímto ovládá přepínání odesílání dat z akcelerometru nebo elektronického gyroskopu.



7.2 Program pro převodník u PC karty

Úkolem tohoto programu je data z UART komunikace převést na SPI sběrnici pro digitálně-analogové převodníky. Dále je jeho úkolem snímat data ze vstupů pro dva zpětné signály a přepínač snímačů. Kompletní zdrojový kód je v příloze F (s. 63). Tento program je též popsán v angličtině.

Nejdůležitější částí programu je obsluha přerušení od přijatých dat. Datová komunikace může vypadnout. Aby se tedy zabránilo pádu výstupního napětí do nulové hodnoty, jsou přijatá data skladována v globálních proměnných. Po přijetí se dělí nejdříve podle toho, zda se jedná o první nebo druhou polovinu dat. Dále se rozdělí podle příslušnosti k první nebo druhé dvojici senzorů – tímto je určen výstupní převodník. Poslední dělení určuje, pro který výstup daného převodníku jsou data určena. Celé toto třídění je obsaženo v obsluze přerušení od přijatých dat UART modulem. Na obr. 7.3 je vývojový diagram obsluhy přerušení.



Obr. 7.3: Vývojový diagram obsluhy přerušení u převodníku pro kartu

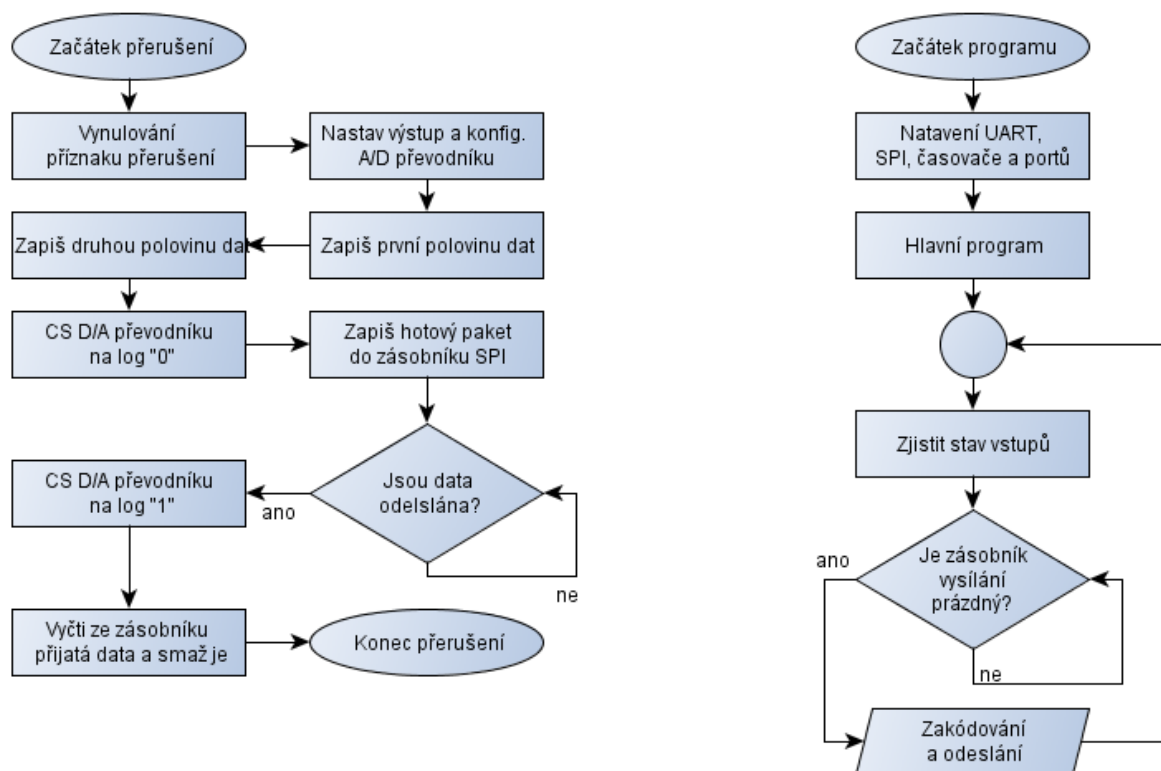


Snímače odpovídají s výstupy převodníku nejjednodušším způsobem – první měřený snímač je přenesen na první výstup prvního převodníku, a tak dále. Propojení snímačů s převodníky ukazuje tab. 7.2.

Tab. 7.2: Vztah snímačů a výstupů D/A převodníků

Snímač	Kódování		D/A převodník a výstup	
Hallův snímač	0	0	První	A
Osa akcelerometru X	0	1		B
Osa akcelerometru Y	1	0	Druhý	A
Osa akc. Z nebo gyroskop	1	1		B

Hlavní program jen dokola kontroluje stav zpětných signálů a přepínače snímačů. Pro výstup dat do digitálně-analogových převodníků je v programu nastaven nezávislý časovač, který po nastaveném čase vyvolá přerušení. V tomto přerušení se zkompletují data z jednotlivých globálních proměnných, doplní se jim konfigurační bity digitálně-analogového převodníku a takto hotová data se odešlou pomocí sběrnice SPI. Na obr. 2.1 je vývojový diagram znázorňující hlavní program a obsluhu přerušení od časovače.



Obr. 7.4: Obsluha přerušení a hlavní program převodníku pro kartu



7.3 Rychlost měření a vyhodnocení dat

Ještě bych rád vysvětlil rychlost běhu programu. Mikrokontrolery mají připojen krystalový oscilátor o frekvenci 22,1184 MHz. Tato frekvence se poté dělí na polovinu pro dosažení instrukčního taktu. Jedna instrukce tedy trvá přibližně 90 ns. V tab. 7.3 je přibližně uvedeno kolik času jednotlivé dílčí operace vyžadují.

Tab. 7.3: Časové náročnosti operací s daty

Operace	Čas pro jedno vykonání	Opakování	Celkový čas
A/D převod	4,9 μ s	5	24,5 μ s
Odeslání UART (13 bitů)	14,1 μ s	8	112,8 μ s
Odeslání SPI (16bitů)	1,4 μ s	4	5,6 μ s
Časovač	0,11 ms		

Jak je zřejmé, nejpomalejší na celém přesunu dat od snímačů k měřicí kartě je UART komunikace. Její přenosová rychlost je nastavena na 921,6 kbps, tedy maximální rychlost jakou dokáže Bluetooth modul komunikovat. Navíc ještě přenáší největší množství dat. Analogovo-digitální převodník by mohl pracovat rychleji, ale vzhledem k možnému zkreslení dat, to není doporučeno. Sběrnice SPI je nastavena na stejnou frekvenci, jakou mají instrukce (dělíčka 1:1) – tedy přibližně 11 MHz. Zde již stejně jako v případě Bluetooth modulu zrychlit nelze. Sečtením těchto časů však výsledný nedostaneme. Je to z důvodu nemalého množství instrukcí podílejících se na zpracování dat. Přesný počet instrukcí potřebných k vykonání programu v letadle je 582 (pozor – instrukcí jazyku Assembler ne C) a program převodníku pro kartu má 690 instrukcí. Z tohoto počtu to ale také říci nelze. A to z toho důvodu, že v tomto čísle jsou zahrnuty i instrukce přípravy převedení jazyka C na Assembler. Pro přesný čas by bylo nutné vyvolat skok napětí na vstupu A/D převodníku a změřit odezvu na výstupu operačního zesilovače. Tento čas se však může měnit spolu s okolnostmi přenosu signálu vzduchem. Navíc tento program není nastaven na co nejrychlejší odezvu, ale na držení stálé obnovovací frekvence na výstupu. Ta je dána nastavením časovače na přibližně 9 kHz.

Pro úlohu je to možná zbytečně rychlé nastavení UART komunikace a obnovovací frekvence. Ale tímto bych rád demonstroval, kam až je možné s podobným řešením zajít.



8 Měření na hotové úloze

Na hotové a zabudované úloze jsem provedl několik měření, které měly za úkol ověřit její funkčnost. Měření probíhalo pomocí měřicí karty v počítači a programu MATLAB. První, co mě při měření překvapilo, byla míra šumu na signálu. Grafy jsem zvětšil, aby měly co největší vypovídací hodnotu, ale tím jsem je musel přesunout do příloh. Následují komentáře k jednotlivým grafům v příloze.

Příloha G (s. 67): zde jsem měřil odezvu na skok výkonu z nuly do maxima a zase zpět. Na začátku jsem nechal motor protáčet, ale jen natolik aby nepohyboval s letadlem. Je tak vidět rušení sršením komutátoru v motoru a slabé vibrace. Po nastavení plného výkonu se vibrace extrémně zvyšují a zakrývají téměř celý průběh měření. Po úplném vypnutí motoru vibrace utichají a tím i rušení na signálech. V tuto chvíli jsou průběhy téměř čisté a vypovídají o setrvačných silách při doběhu letadla.

Příloha H (s. 68): graf ukazuje odezvu úlohy na lineární vzestup napájecího napětí. Po odhalení vibrací jakožto hlavního zdroje rušení jsem změřil závislost měřených vibrací na výkonu (a tím i otáčkách) motoru. Z charakteristiky je zřejmé, že při různých otáčkách jsou vibrace velice různé. Z těchto údajů o rezonančních frekvencích by se poté i dala přibližně určit váha rezonujících částí na letadle.

Příloha I (s. 69): abych mohl s jistotou říci, že šum vyvolávají vibrace a ne přenos napájecího napětí přes kroužky. Zapnul jsem motor opět do volnoběžných otáček, podržel úlohu rukou a po uplynutí nastaveného času ovládací program sám nastavil maximální výkon. To je na grafu vidět silným rušením, ale bez snížení napětí na Hallově snímači. Poté jsem maketu pustil, načež se postupně rozeběhla. Toto je již doprovázeno změnou napětí Hallova snímače. Při ožívování zapojení také nebylo zjištěno žádné vlastní rušení signálu modulů, jen čisté průběhy.

Ze všech průběhů je také zřejmá limitace Hallova snímače. Ta může být způsobena jeho „mrtvou zónou“, která začíná kousek od krajní polohy. Nebo je možné, že je toto zcela normální chování, protože motor nemá dostatečný výkon pro další vyzdvižení. Vzhledem k neznalosti těchto problémů v průběhu úprav a jejich povaze (umístění snímače, uložení motoru) již poté nebylo místo pro jejich řešení. Řešení podobných problémů nebylo úkolem této práce. Rušení od vibrací by se dalo odstranit elektronicky pomocí filtru, avšak tento filtr by již mohl tlumit rychlejší pohyby celé úlohy a tím je zkreslit. Nebo mechanicky jiným uložením pohonu letadla.



9 Závěr

Seznámil jsem se s různými možnostmi přenosu dat pro laboratorní úlohu. Výběrovými kritérii byla zejména rychlost přenosu dat, robustnost přenosu a v neposlední řadě cena a dostupnost. Zvolil jsem tedy způsob přenosu pomocí Bluetooth. Dále jsem zvážil možnosti řešení modulu pro odměřování. Kritériem výběru byla v první řadě cena, dále pak univerzálnost a snadná opravitelnost. Jako optimální variantu jsem zvolil řešení s mikrokontrolerem PIC24F04KA201 a komunikačním modulem OEMSPA310i. Vzhledem k zadání bylo zapojení doplněno o akcelerometr a gyroskop. Tyto senzory však nejsou hlavním zdrojem údajů o poloze, tím je absolutní snímač polohy využívající principu Hallova jevu. S pomocí prototypu složeného z hlavních prvků obvodu, jsem v jazyku C vypracoval program řídící jednotlivé operace. Odladěním programu jsem postoupil k výrobě a osazení výsledných oboustranných desek plošných spojů. Díky bezdrátovému přenosu signálu se mi podařilo, pro pozdější rozšíření nebo úpravu, vést do modulu pro letadlo také dva prozatím nevyužité datové signály. Do laboratorní úlohy jsem moduly zabudoval s ohledem na možné rušení. Při měření se však ukázaly problémy s celkovou mechanickou konstrukcí úlohy. Jedná se o elektromagnetické rušení od silových vodičů s pulsně šířkovou modulací a přenos signálů pomocí kroužků. Hlavním zdrojem rušení však jsou vibrace způsobené motorem. Ty značně ovlivňují desku snímačů i přes její pružné uložení. Vibrace však ovlivňují pouze signály z inerciálních snímačů, ne však hlavní snímač polohy. Odstranění rušení způsobené vibracemi a silovým vedením nebylo úkolem této práce. Mohlo by však být námětem pro další bakalářskou nebo diplomovou práci. Hlavní odměřování polohy je plně funkční.

Zvolenou vysokou rychlostí obnovy dat jsem demonstroval možné další uplatnění modulů. Příkladem může být jejich použití pro odměřování pohybu robotických ramen nebo vozíků, kde by rušení vibracemi mohlo být menší. Díky unifikovanému měřicímu signálu o hodnotě 0 – 10V je možné tyto moduly použít bez jakýchkoliv úprav i v jiných aplikacích.



Seznam použité literatury

- [1] *Český telekomunikační úřad* [online]. Praha : 2004 [cit. 2011-05-08]. Plán přidělení kmitočtových pásem. Dostupné z WWW: <http://www.ctu.cz/1/download/plan-prideleni-kmitoctovych-pasem_1114099610.pdf>.
- [2] *IEEE802.15* [online]. 2010-01-10 [cit. 2011-05-08]. IEEE 802.15 WPAN™ Task Group 4 (TG4). Dostupné z WWW: <<http://www.ieee802.org/15/pub/TG4.html>>.
- [3] *ZigBee Alliance* [online]. 2011 [cit. 2011-05-08]. Dostupné z WWW: <www.zigbee.org>.
- [4] ŘEHÁK, Jan. *HW.cz* [online]. 2003-06-21 [cit. 2011-05-08]. Co je to WiFi - úvod do technologie. Dostupné z WWW: <<http://hw.cz/Produkty/Ethernet/ART915-Co-je-to-WiFi---uvod-do-technologie.html>>.
- [5] Bluetooth SIG, Inc. *Bluetooth* [online]. c2011 [cit. 2011-05-08]. Dostupné z WWW: <<http://www.bluetooth.com/Pages/Bluetooth-Home.aspx>>.
- [6] *Freescale* [online]. c2007 [cit. 2011-05-09]. Wireless Sensing Triple-Axis Reference Desing. Dostupné z WWW: <http://cache.freescale.com/files/sensors/doc/fact_sheet/ZSTARBOARDFS.pdf?fpsp=1>.
- [7] *ConnectBlue* [online]. c2011 [cit. 2011-05-10]. Wireless LAN (WLAN) Serial Port Adapter Module - OWS451 (UART interface). Dostupné z WWW: <<http://www.connectblue.com/products/wireless-lan-products/wireless-lan-modules/wireless-lan-uart-serial-port-module-ows451/>>.
- [8] *Radiocrafts : Embedded Wirless Solutions* [online]. c2011 [cit. 2011-05-09]. Dostupné z WWW: <<http://www.radiocrafts.com/index.php?sideID=127>>.
- [9] ŠKUREK, Pavel. *HW.cz* [online]. 2005-06-21 [cit. 2011-05-09]. RC220x - komunikační moduly pro Zigbee. Dostupné z WWW: <<http://hw.cz/Firemni-clanky/Macroweil/ART1316-RC220x---komunikacni-moduly-pro-Zigbee.html>>.
- [10] *Microchip* [online]. c2009, Last Updated: 06/02/2009 [cit. 2011-05-11]. Datasheet PIC24F04KA201. Dostupné z WWW: <<http://ww1.microchip.com/downloads/en/DeviceDoc/39937b.pdf>>.
- [11] *Microchip* [online]. c2006, Last Updated: 03/28/2006 [cit. 2011-05-11]. PIC18F to PIC24F Migration: An Overview. Dostupné z WWW: <<http://ww1.microchip.com/downloads/en/DeviceDoc/39764a.pdf>>.
- [12] *Microchip* [online]. 2006, Last Updated: 09/21/2009 [cit. 2011-05-11]. PIC24F Family Reference Manual, Sect. 17 10-Bit A/D Converter. Dostupné z WWW: <<http://ww1.microchip.com/downloads/en/DeviceDoc/39705b.pdf>>.



- [13] *Spezial elektronik* [online]. 2008-05 [cit. 2011-05-13]. Datasheet OEMSPA310i. Dostupné z WWW: <http://www.spezial.cz/pdf/em_ds_oemspa_310.pdf>.
- [14] *Spezial elektronik* [online]. c2011 [cit. 2011-05-13]. Univerzální dálkové ovládání s Bluetooth. Dostupné z WWW: <<http://www.spezial.cz/apps/dalkove-ovladani-rc-bluetooth.html>>.
- [15] *Freescale Semiconductor* [online]. 2005, Rev 5, 03/2008 [cit. 2011-05-13]. Datasheet MMA7260Q. Dostupné z WWW: <http://cache.freescale.com/files/sensors/doc/data_sheet/MMA7260QT.pdf?fsrch=1&sr=1>.
- [16] *InvenSense* [online]. 10/28/08, 05/20/10 [cit. 2011-05-13]. Datasheet ISZ-500. Dostupné z WWW: <<http://invensense.com/mems/gyro/documents/ps-isz-0500b-00-06.pdf>>.
- [17] *Spezial elektronik* [online]. c2011 [cit. 2011-05-13]. 3,3V UART<>RS232 převodník s UM3232. Dostupné z WWW: <http://www.spezial.cz/apps/3V-uart-rs232-prevodnik_um3232.html>.
- [18] *MAXIM* [online]. 2006, 07-2010 [cit. 2011-05-13]. Datasheet MAX220 - MAX249. Dostupné z WWW: <<http://datasheets.maxim-ic.com/en/ds/MAX220-MAX249.pdf>>.
- [19] *Spezial elektronik* [online]. c2011 [cit. 2011-05-14]. Bezdrátová RS232 Bluetooth / WiFi náhrada sériového kabelu. Dostupné z WWW: <<http://www.spezial.cz/connectblue/bezdratova-bluetooth-seriova-linka-rs232.html>>.
- [20] *Microchip* [online]. 2010 [cit. 2011-05-15]. Datasheet MCP4902/4912/4922. Dostupné z WWW: <<http://ww1.microchip.com/downloads/en/DeviceDoc/22250A.pdf>>.
- [21] *Analog Devices* [online]. c2011 [cit. 2011-05-15]. Datasheet AD820. Dostupné z WWW: <http://www.analog.com/static/imported-files/data_sheets/AD820.pdf>.



Seznam příloh na CD

Datové listy součástek

Měření na letadle

Návrh desek

Bezdrátový přenos signálu v reálné aplikaci na letadle.docx

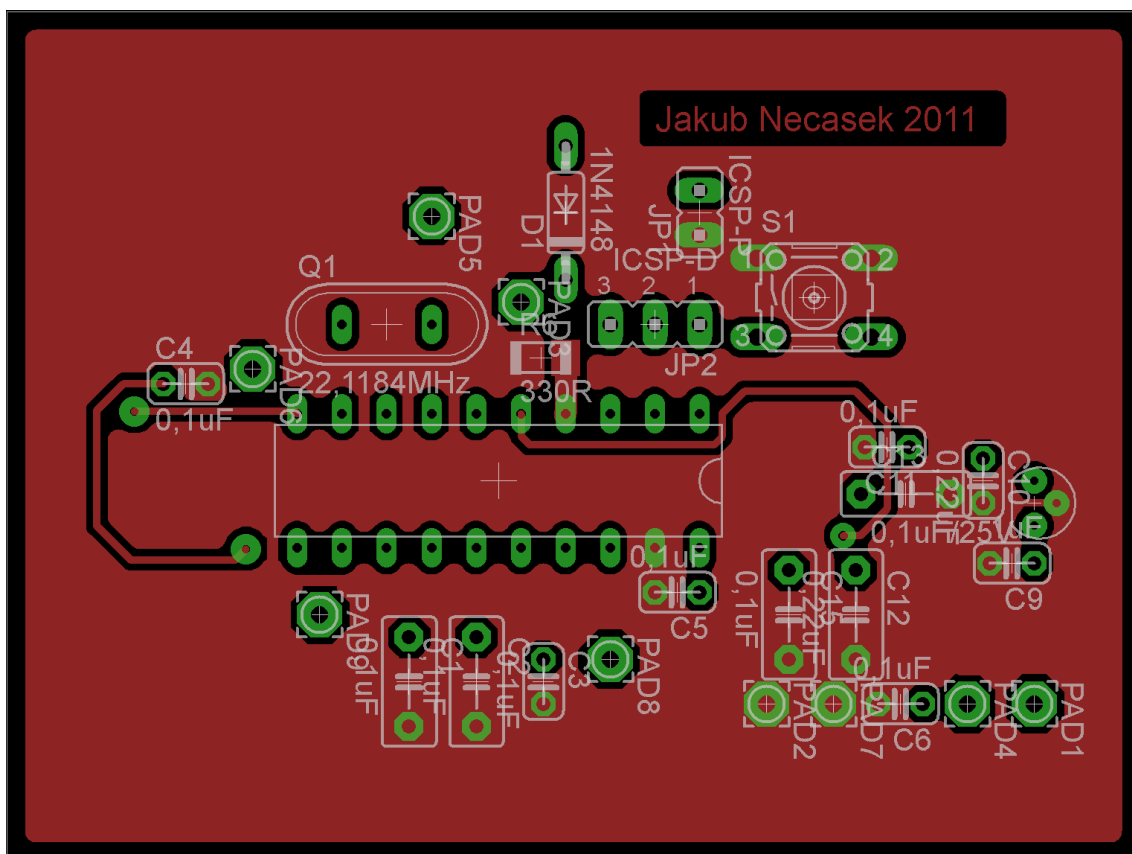
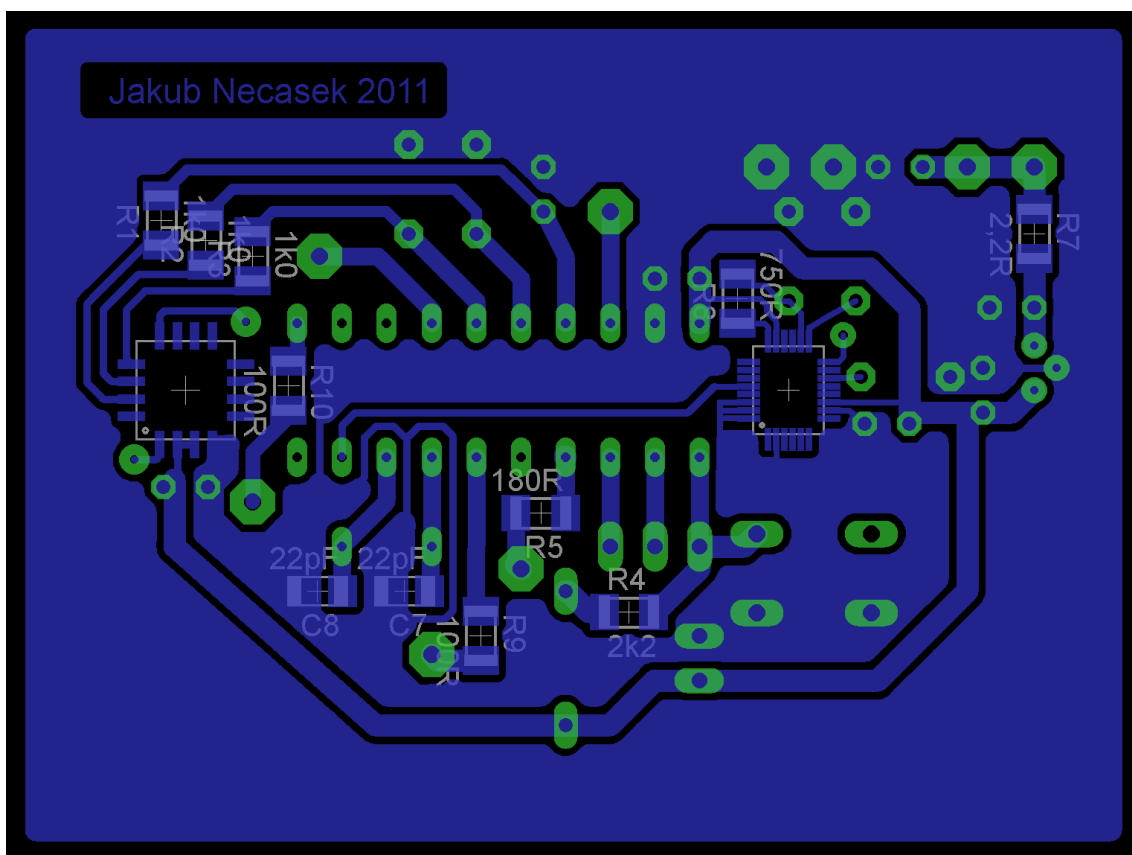
Bezdrátový přenos signálu v reálné aplikaci na letadle.pdf

Program_strana_letadlo.c

Program_strana_PC.c

SPA_Toolbox_v3.10.zip

Příloha A

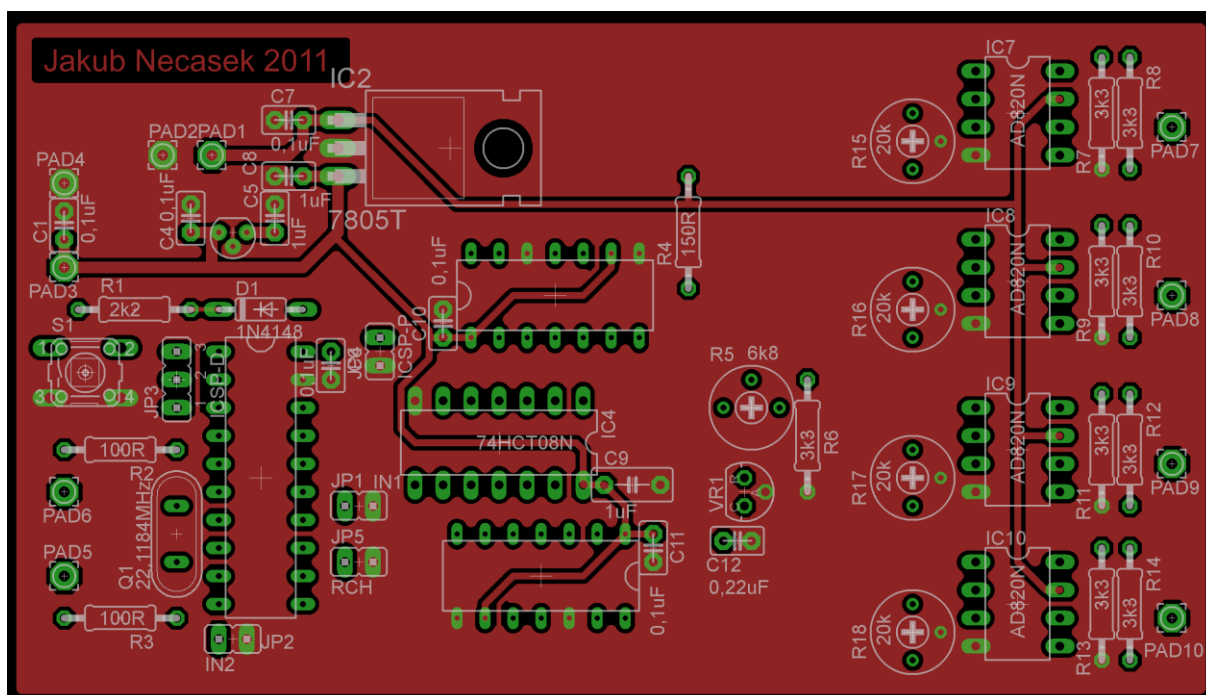
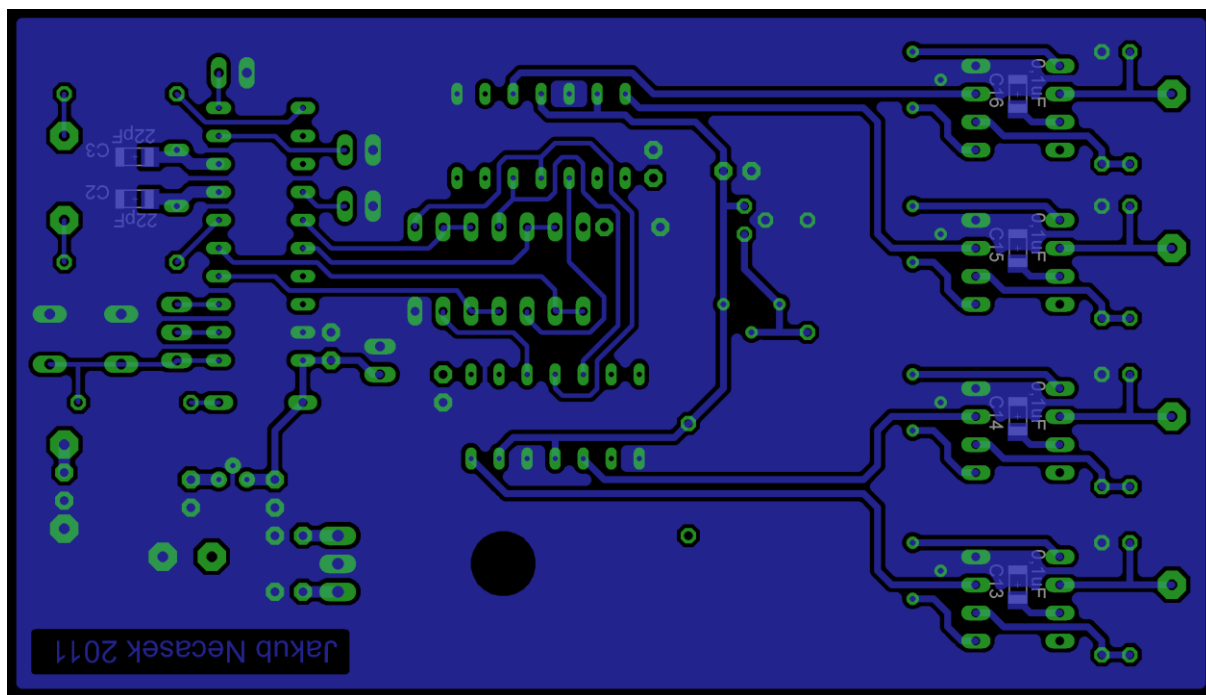




Příloha B

Součástka	Hodnota	Pouzdro
C1	0,1uF	C5B3
C2	0,1uF	C5B3
C3	0,1uF	C025-024X044
C4	0,1uF	C025-024X044
C5	0,1uF	C025-024X044
C6	0,1uF	C025-024X044
C7	22pF	C1206
C8	22pF	C1206
C9	1uF	C025-024X044
C10	0,22uF	C025-024X044
C11	0,1uF	C025-024X044
C12	0,22uF	C5B3
C13	0,1uF/25V	C5B2.5
C15	0,1uF	C5B3
D1	1N4148	DO35-7
IC1	TSC295033	TO92
IC2	PIC24F04KA201	PDIP20
IC3	MMA7260Q	QFN16
IC4	ISZ-500	QFN28
JP1	ICSP-P	JP1
JP2	ICSP-D	JP2
PAD1	2,54/1,0	
PAD2	2,54/1,0	
PAD3	2,54/1,0	
PAD4	2,54/1,0	
PAD5	2,54/1,0	
PAD6	2,54/1,0	
PAD7	2,54/1,0	
PAD8	2,54/1,0	
PAD9	2,54/1,0	
Q1	22,1184MHz	QS
R1	1k0	M1206
R2	1k0	M1206
R3	1k0	M1206
R4	2k2	M1206
R5	180R	M1206
R6	330R	M1206
R7	2,2R	M1206
R8	750R	M1206
R9	100R	M1206
R10	100R	M1206
S1		B3F-10XX

Příloha C





Příloha D

Součástka	Hodnota	Pouzdro	Součástka	Hodnota	Pouzdro
C1	0,1uF	C2.5-2	JP5	RCH	JP1
C2	22pF	C1206	PAD1	2,54/1,0	
C3	22pF	C1206	PAD2	2,54/1,0	
C4	0,1uF	C2.5-2	PAD3	2,54/1,0	
C5	1uF	C2.5-2	PAD4	2,54/1,0	
C6	0,1uF	C2.5-2	PAD5	2,54/1,0	
C7	0,1uF	C2.5-2	PAD6	2,54/1,0	
C8	1uF	C2.5-2	PAD7	2,54/1,0	
C9	1uF	C5B3	PAD8	2,54/1,0	
C10	0,1uF	C2.5-2	PAD9	2,54/1,0	
C11	0,1uF	C2.5-2	PAD10	2,54/1,0	
C12	0,22uF	C2.5-2	Q1	22,1184MHz	QS
C13	0,1uF	C1206	R1	2k2	0207/10
C14	0,1uF	C1206	R2	100R	0207/10
C15	0,1uF	C1206	R3	100R	0207/10
C16	0,1uF	C1206	R4	150R	0207/10
D1	1N4148	DO35-7	R5	6k8	RTRIM3339P
IC1	TSC295033	TO92	R6	3k3	0207/10
IC2	7805T	TO220H	R7	3k3	0207/10
IC3	PIC24F04KA201	PDIP20	R8	3k3	0207/10
IC4	74HCT08N	DIL1	R9	3k3	0207/10
IC5	MCP4922	PDIP	R10	3k3	0207/10
IC6	MCP4922	PDIP	R11	3k3	0207/10
IC7	AD820N	DIL08	R12	3k3	0207/10
IC8	AD820N	DIL08	R13	3k3	0207/10
IC9	AD820N	DIL08	R14	3k3	0207/10
IC10	AD820N	DIL0	R15	20k	RTRIM3339P
JP1	IN1	JP1	R16	20k	RTRIM3339P
JP2	IN2	JP1	R17	20k	RTRIM3339P
JP3	ICSP-D	JP2	R18	20k	RTRIM3339P
JP4	ICSP-P	JP1	S1		B3F-10XX
			VR1	TL431	TO92-CLP



Příloha E

```
#include <p24f04ka201.h>

_FOSCSEL( IESO_OFF & FNOSC_PRI )
_FOSC( FCKSM_CSDCMD & SOSCSSEL_SOSCLP & POSCFREQ_HS & OSCIOFNC_ON &
POSCMOD_HS )
_FWDT( FWDTEN_OFF )
_FPOR( MCLRE_ON & BORV_27V & PWRTEN_OFF & BOREN_BOR3 )
_FICD( ICS_PGx2 )
_FDS( DSWDTEN_OFF & DSLPBOR_OFF & DSWCKSEL_LPRC & DSWDTPS_DSWDTPS6 )

unsigned short HALL;
unsigned short ACX;
unsigned short ACY;
unsigned short ACZ;
unsigned short GYRO;
unsigned char SELECT;

void main()                //main program
{

    U1BRG = 0x0002;        //Set Baudrate 921,6kbps
    IPC2bits.U1RXIP2 = 1;  //Set UART RX Interrupt Priority
    IPC2bits.U1RXIP1 = 0;
    IPC2bits.U1RXIP0 = 1;
    U1STA = 0;
    U1MODE = 0xA00D;       //set UART: 8-bit data, odd parity, 2 STOP bits,
                           rts&cts off
    U1STAbits.UTXEN = 1;   //Enable Transmit
    IEC0bits.U1RXIE = 1;   //Enable Receive Interrupt

    AD1PCFG = 0x0033;      //select I/O or AD inputs
    AD1CON2 = 0x0410;      //Vdd&Vss, interrupt after 5 samples
    AD1CON3 = 0x0F02;      //Tad=2*Tcy, Tsam=15*Tad, Tcon=12*Tad
    AD1CON1 = 0x00E0;      //autoconvert
    AD1CSSL = 0x1C0C;      //scan all 5 inputs
    AD1CHS = 0;            //negative input is ground
    IFS0bits.AD1IF = 0;    //clear ADC interrupt flag
    AD1CON1bits.ADON = 1;  //enable ADC

    TRISAbits.TRISA6 = 0;  //port A6 is output
    TRISAbits.TRISA4 = 0;  //port A4 is output
    TRISBbits.TRISB4 = 0;  //port B4 is output
    TRISBbits.TRISB15 = 0; //port B15 is output
    LATBbits.LATB4 = 1;    //disable gyro autozero
    LATAbits.LATA4 = 0;    //akcelerometr at 4g resolution

    unsigned short TEMP;
    SELECT = 0x00;

    while(1)
    {
        AD1CON1bits.ASAM = 1;        //start auto conversion
        while (IFS0bits.AD1IF == '\0') {}; //wait for interrupt flag
        AD1CON1bits.ASAM = 0;        //stop auto conversion
        IFS0bits.AD1IF = 0;          //clear ADC interrupt flag
    }
}
```



```
HALL = ADC1BUF0;           //copy results from ADC buffer
GYRO = ADC1BUF1;
ACX = ADC1BUF2;
ACY = ADC1BUF3;
ACZ = ADC1BUF4;

while(U1STAbits.UTXBF == 1 ) {}
TEMP = HALL;
TEMP &= 0x03E0;
TEMP >>= 5;
TEMP |= 0x0080;
U1TXREG = (unsigned char) TEMP;

while(U1STAbits.UTXBF == 1 ) {}
TEMP = HALL;
TEMP &= 0x001F;
U1TXREG = (unsigned char) TEMP;

while(U1STAbits.UTXBF == 1 ) {} //while transmit buffer is empty
TEMP = ACX;                     //copy data to temporary variable
TEMP &= 0x03E0;                 //upper data mask
TEMP >>= 5;                     //rotate by 5 to the right
TEMP |= 0x00A0;                 //add address- 1st DAC, 2nd output, start
U1TXREG = (unsigned char) TEMP; //send it

while(U1STAbits.UTXBF == 1 ) {} //while transmit buffer is empty
TEMP = ACX;                     //copy data to temporary variable
TEMP &= 0x001F;                 //lower data mask
TEMP |= 0x0020;                 //add address- 1st DAC, 2nd output, end
U1TXREG = (unsigned char) TEMP; //send it

while(U1STAbits.UTXBF == 1 ) {}
TEMP = ACY;
TEMP &= 0x03E0;
TEMP >>= 5;
TEMP |= 0x00C0;
U1TXREG = (unsigned char) TEMP;

while(U1STAbits.UTXBF == 1 ) {}
TEMP = ACY;
TEMP &= 0x001F;
TEMP |= 0x0040;
U1TXREG = (unsigned char) TEMP;

if (SELECT == 0x00)
{
while(U1STAbits.UTXBF == 1 ) {}
TEMP = ACZ;
TEMP &= 0x03E0;
TEMP >>= 5;
TEMP |= 0x00E0;
U1TXREG = (unsigned char) TEMP;

while(U1STAbits.UTXBF == 1 ) {}
TEMP = ACZ;
TEMP &= 0x001F;
TEMP |= 0x0060;
U1TXREG = (unsigned char) TEMP;
}
```



```
if (SELECT == 0xFF)
{
    while(U1STAbits.UTXBF == 1 ) {};
    TEMP = GYRO;
    TEMP &= 0x03E0;
    TEMP >>= 5;
    TEMP |= 0x00E0;
    U1TXREG = (unsigned char) TEMP;

    while(U1STAbits.UTXBF == 1 ) {};
    TEMP = GYRO;
    TEMP &= 0x001F;
    TEMP |= 0x0060;
    U1TXREG = (unsigned char) TEMP;
}

}

void __attribute__((__interrupt__)) _U1RXInterrupt(void) //recieving int.
{
    IFS0bits.U1RXIF = 0; //clear recieve interrupt flag
    unsigned char TEMP;
    while(U1STAbits.URXDA == 1) //while recieve buffer is empty
    {
        TEMP = U1RXREG; //copy data to temporary variable
        if (TEMP == 0x11) //if 11h
        {
            LATAbits.LATA6 = 1; //set port A6
        }

        if (TEMP == 0x01) //if 1h
        {
            LATAbits.LATA6 = 0; //reset port A6
        }

        if (TEMP == 0x12)
        {
            LATBbits.LATB15 = 1;
        }

        if (TEMP == 0x02)
        {
            LATBbits.LATB15 = 0;
        }

        if (TEMP == 0x13)
        {
            SELECT = 0xFF;
        }

        if (TEMP == 0x03)
        {
            SELECT = 0x00;
        }
    }
}
```



Příloha F

```
#include <p24f04ka201.h>

_FOSCSEL( IESO_OFF & FNOSC_PRI )
_FOSC( FCKSM_CSDCMD & SOSCSEL_SOSCLP & POSCFREQ_HS & OSCIOFNC_ON &
POSCMOD_HS )
_FWDT( FWDTEN_OFF )
_FPOR( MCLRE_ON & BORV_27V & PWRTEN_OFF & BOREN_BOR3 )
_FICD( ICS_PGx2 )
_FDS( DSWDTEN_OFF & DSLPBOR_OFF & DSWCKSEL_LPRC & DSWDTPS_DSWDTPS6 )

unsigned char A00;
unsigned char A01;
unsigned char A10;
unsigned char A11;
unsigned char B00;
unsigned char B01;
unsigned char B10;
unsigned char B11;

void main()
{
AD1PCFG = 0xFFFF;           //enable I/O ports instead ADC input
AD1CON1bits.ADON = 0;       //ADC off

U1BRG = 0x0002;             //Set Baudrate 921,6kbps
IPC2bits.U1RXIP2 = 1;       //Set Uart RX interrupt priority
IPC2bits.U1RXIP1 = 0;
IPC2bits.U1RXIP0 = 1;
U1STA = 0;
U1MODE = 0xA00D;           //Enable Uart for 8-bit data, odd parity, 2
                             //STOP bits
U1STAbits.UTXEN = 1;       //Enable Transmit
IEC0bits.U1RXIE = 1;       //Enable Receive interrupt

SPI1CON1 = 0x053F;         //MSTR, SDO&SCK on, !SS1 off, falling edge,
                             //active = 1, 11MHz
SPI1CON2 = 0;
SPI1STATbits.SPIROV = 0;
SPI1STATbits.SPIEN = 1;    //Enable SPI

T1CON = 0x00;              //Stops the Timer1 and reset control reg.
TMR1 = 0x00;               //Clear contents of the timer register
PR1 = 0x04CD;              //Interrupt rate = 9kHz (aprox)
IPC0bits.T1IP2 = 1;        //Set Timer1 interrupt priority
IPC0bits.T1IP1 = 1;
IPC0bits.T1IP0 = 0;
IFS0bits.T1IF = 0;         //Clear the Timer1 interrupt status flag
IEC0bits.T1IE = 1;         //Enable Timer1 interrupts
T1CONbits.TON = 1;         //Start Timer1, 1:1, internal osc

TRISA = 0x00F3;            //Set directon of ports A
TRISB = 0x0C78;            //Set directon of ports B
CNPU1 = 0x0103;            //enable pull-up resistors
LATBbits.LATB0 = 1;
LATBbits.LATB1 = 1;
```



```
while(1)
{
    if (PORTAbits.RA6 == '\0' && U1STAbits.UTXBF == '\0')
    {
        //if port A6 is low and buffer is empty
        U1TXREG = 0x11;    //send 11h
    }

    if (PORTAbits.RA6 == 1 && U1STAbits.UTXBF == '\0')
    {
        U1TXREG = 0x01;
    }

    if (PORTAbits.RA4 == '\0' && U1STAbits.UTXBF == '\0')
    {
        U1TXREG = 0x12;
    }

    if (PORTAbits.RA4 == 1 && U1STAbits.UTXBF == '\0')
    {
        U1TXREG = 0x02;
    }

    if (PORTBbits.RB4 == '\0' && U1STAbits.UTXBF == '\0')
    {
        U1TXREG = 0x13;
    }

    if (PORTBbits.RB4 == 1 && U1STAbits.UTXBF == '\0')
    {
        U1TXREG = 0x03;
    }
}

void __attribute__((__interrupt__)) _U1RXInterrupt(void)
{
    IFS0bits.U1RXIF = 0;    //Clear the UART1 interrupt status flag
    LATBbits.LATB0 = 1;
    LATBbits.LATB1 = 1;
    unsigned char URXRD;
    unsigned char URXRA;
    while(U1STAbits.URXDA == 1)
    {
        URXRD = U1RXREG;
        URXRA = URXRD & 0x60;
        if (URXRD > 0x7F) //begin of data packet?
        {
            if (URXRA < 0x40) //first DAC?
            {
                switch (URXRA)    //which output?
                {
                    case 0x00 :    //1st DAC, 1st output
                        URXRD &= 0x1F;    //delete address
                        A00 = URXRD;    //write first packet
                        break;

                    case 0x20 :    //1st DAC, 2nd output
                        URXRD &= 0x1F;    //delete address
                        A10 = URXRD;    //write first packet
                        break;
                }
            }
        }
    }
}
```




```

    }
    if (URXRA > 0x2F) //second DAC?
    {
        switch (URXRA)    //which output?
        {
            case 0x40 :    //2nd DAC, 1st output
                URXRD &= 0x1F;    //delete adress
                B00 = URXRD;    //write first packet
                break;

            case 0x60 :    //2nd DAC, 2nd output
                URXRD &= 0x1F;    //delete adress
                B10 = URXRD;    //write first packet
                break;

        }
    }
}
if (URXRD < 0x80) //end of data packet?
{
    if (URXRA < 0x40) //first DAC?
    {
        switch (URXRA)    //which output?
        {
            case 0x00 :    //1st DAC, 1st output
                URXRD &= 0x1F;    //delete adress
                A01 = URXRD;    //write second packet
                break;

            case 0x20 :    //1st DAC, 2nd output
                URXRD &= 0x1F;    //delete adress
                A11 = URXRD;    //write second packet
                break;

        }
    }
    if (URXRA > 0x2F) //second DAC?
    {
        switch (URXRA)    //which output?
        {
            case 0x40 :    //2nd DAC, 1st output
                URXRD &= 0x1F;    //delete adress
                B01 = URXRD;    //write second packet
                break;

            case 0x60 :    //2nd DAC, 2nd output
                URXRD &= 0x1F;    //delete adress
                B11 = URXRD;    //write second packet
                break;

        }
    }
}
}

void __attribute__((__interrupt__)) _T1Interrupt(void)
{
    IFS0bits.T1IF = 0;    //Clear the Timer1 interrupt status flag
    LATBbits.LATB0 = 1;
    LATBbits.LATB1 = 1;
    unsigned short TEMP;
    unsigned short TSPI;
}

```



```
TSPI = 0x7000; //write adress - first output
TEMP = (unsigned short) A00; //read first half of data
TEMP <=<= 7;
TSPI |= TEMP; //write first packet
TEMP = (unsigned short) A01; //read second half of data
TEMP <=<= 2;
TSPI |= TEMP; //write second packet

LATBbits.LATB0 = 0; //first DAC CS low
SPI1BUF = TSPI; //send to DAC
while(SPI1STATbits.SPIRBF == '\0') {}; //wait to end of transfer
LATBbits.LATB0 = 1; //first DAC CS high
TSPI = SPI1BUF;

TSPI = 0xF000; //write adress - second output
TEMP = (unsigned short) A10;
TEMP <=<= 7;
TSPI |= TEMP; //write first packet
TEMP = (unsigned short) A11;
TEMP <=<= 2;
TSPI |= TEMP; //write second packet

LATBbits.LATB0 = 0; //first DAC
SPI1BUF = TSPI; //send to DAC
while(SPI1STATbits.SPIRBF == '\0') {};
LATBbits.LATB0 = 1;
TSPI = SPI1BUF;

TSPI = 0x7000; //write adress - first output
TEMP = (unsigned short) B00;
TEMP <=<= 7;
TSPI |= TEMP; //write first packet
TEMP = (unsigned short) B01;
TEMP <=<= 2;
TSPI |= TEMP; //write second packet

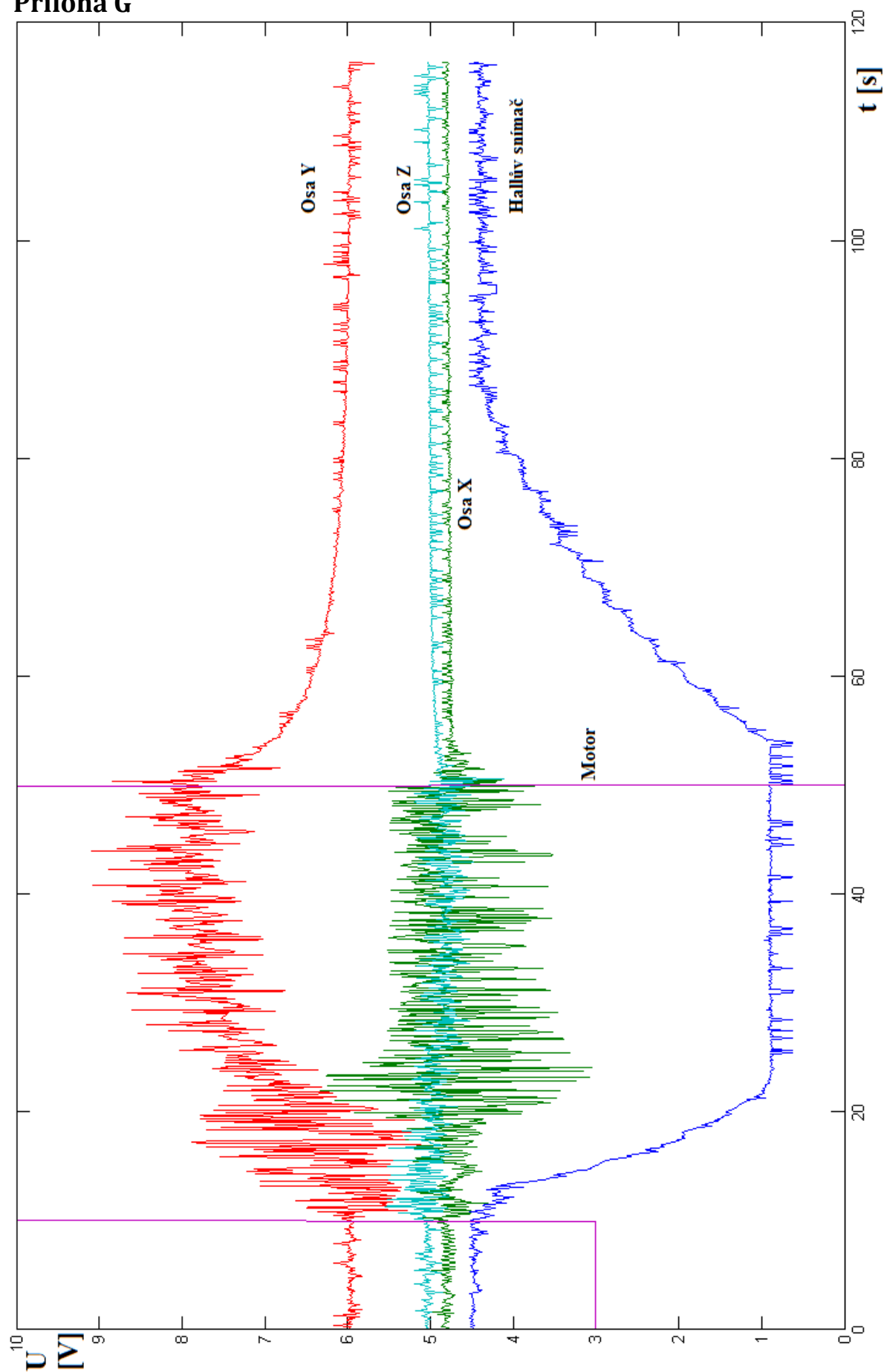
LATBbits.LATB1 = 0; //second DAC
SPI1BUF = TSPI; //send to DAC
while(SPI1STATbits.SPIRBF == '\0') {};
LATBbits.LATB1 = 1;
TSPI = SPI1BUF;

TSPI = 0xF000; //write adress - second output
TEMP = (unsigned short) B10;
TEMP <=<= 7;
TSPI |= TEMP; //write first packet
TEMP = (unsigned short) B11;
TEMP <=<= 2;
TSPI |= TEMP; //write second packet

LATBbits.LATB1 = 0; //second DAC
SPI1BUF = TSPI; //send to DAC
while(SPI1STATbits.SPIRBF == '\0') {};
LATBbits.LATB1 = 1;
TSPI = SPI1BUF;
}
```

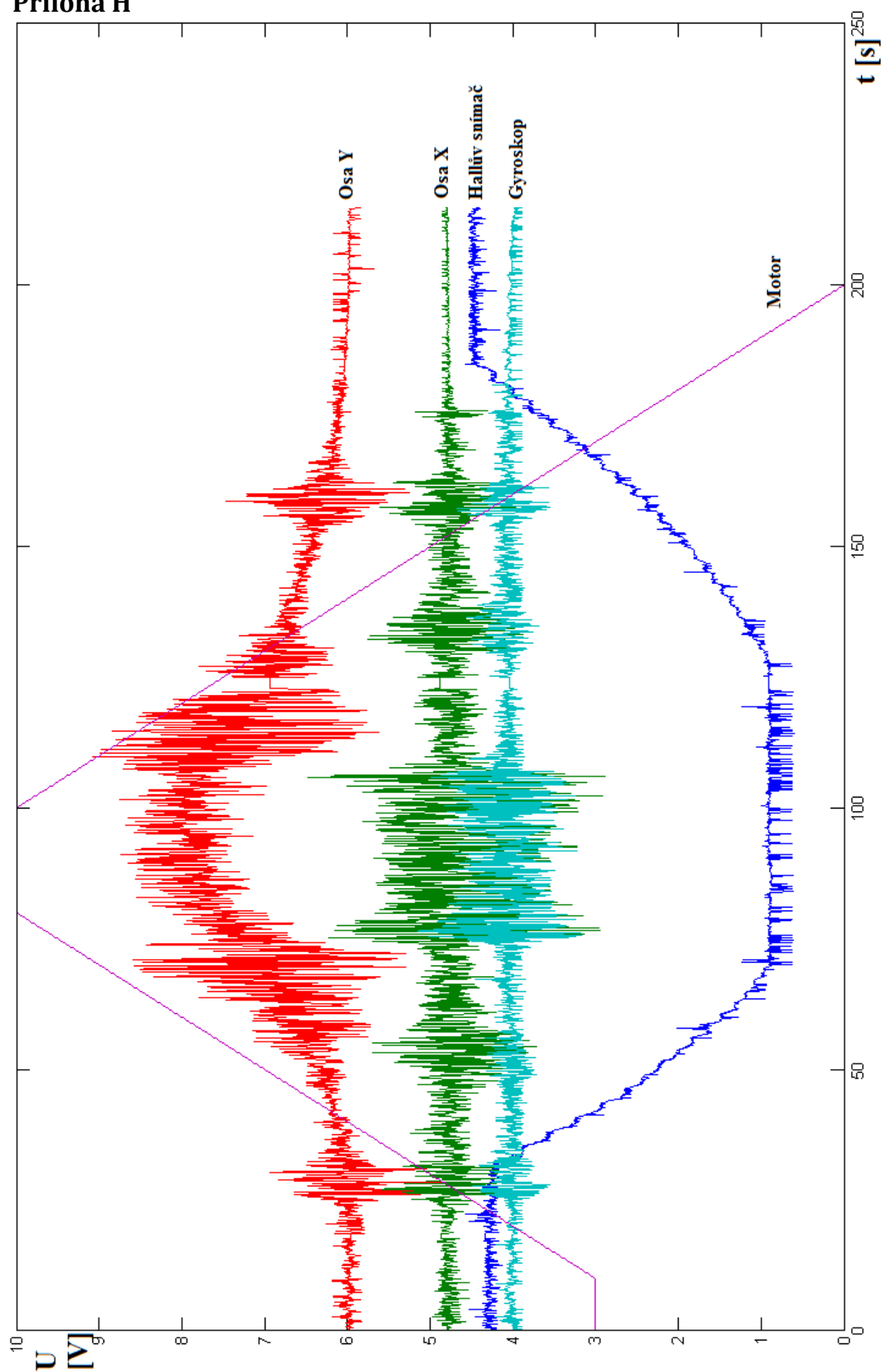


Příloha G





Příloha H





Příloha I

